5

Z MIKROKOMPUTEREM NA TY

Bajtek

NR INDEKSU 353965
PL ISSN 0860-1674

MIESIĘCZNY DODATEK DO SZTANDARU MŁODYCH NR 5(41) MAJ 1989 CENA 220 ZŁ

NOWY KONKURS!

STRUKTURY DANYCH

MicroDraft

MONITORY

CeBIT'89

OKNO NA ŚWIAT

POLICE QUEST

SM SZTANDAR MŁODYCH

NAUKOWE SZALEŃSTWO

Dotyczy ono oczywiście zainicjowanego przez profesorów Martina Fleischmanna i Stanleya Ponsa wyścigu czołowych laboratoriów świata do uzyskania syntezy jądrowej w temperaturze pokojowej. W poprzednim numerze pisałem o rewelacji ogłoszonej przez obu profesorów z dużą emfazą, gdyż sprawa, sama w sobie, jest tego z pewnością warta. Jednak kilka tygodni, które upłynęły od ogłoszenia o odkryciu, znacznie ostudziły emocje.

Najpierw zalani zostaliśmy falą informacji z różnych ośrodków naukowych, raczej do tej pory nieznanych, o powtórzeniu eksperymentu Fleischmanna-Ponsa. Jednak tym pierwszym, hurra optymistycznym doniesieniom, towarzyszyło zastanawiające milczenie głównych światowych ośrodków zajmujących się badaniami nad syntezą jądrową. Tajemnica tego milczenia została wkrótce wyjaśniona: trwały tam przygotowania do weryfikacji hipotezy (bo każde odkrycie, dopóki nie zostanie potwierdzone w szeregu niezależnych doświadczeń, jest tylko hipotezą!) sformułowanej w Uniwersytecie Utah.

Po mniej więcej dwóch — trzech tygodniach zaczęły nadchodzić informacje o wynikach tych kontrolnych doświadczeń. Wyniki, niestety, negatywne. Choć gwoli formalności należy powiedzieć, że trudno właściwie mówić o powtórzeniach eksperymentu, gdyż okazało się wkrótce, że Fleischmann i Pons odmówili ujawnienia kluczowych szczegółów swojego osiągnięcia — tak im bowiem poradzili prawnicy wynajęci do zajęcia się opatentowaniem i wyciągnięciem „szmalu" z tego odkrycia.

Fleischmann i Pons znaleźli się w opałach. Najsilniejszy atak przypuścili fizycy, dysponujący do tej pory monopolem na badania w zakresie syntezy jądrowej. Uczestnicy dorocznego zjazdu Amerykańskiego Towarzystwa Fizycznego, który odbył się na początku maja w Baltimore, nie zostawili na autorach kontrowersyjnego eksperymentu suchej nitki. Elektromechanicy zapowiedzieli odwet na swoim zjeździe w Los Angeles...

Ale przecież nie o argumenty słowne w całej tej zabawie chodzi! Najlepszym argumentem „za" byłyby przecież nie budzące zastrzeżeń, udane doświadczenia, możliwe do powtórzenia przez każdego, kto tego zapragnie i ma niezbędne ku temu wyposażenie oraz doświadczenie w przeprowadzaniu eksperymentów. Na razie jednak (piszę te słowa na początku maja) takich przekonujących argumentów brakuje.

Nie zazdroszczę Fleischmannowi i Ponsowi, jeśli okaże się ostatecznie, że nie mieli racji! Jednak jest w tej całej historii, niezależnie od możliwych jej zakończeń, element niezwykle pozytywny. Otóż okazało się przecież, że ich komunikat pobudził do działania tysiące, a może nawet i setki tysięcy naukowców na całym świecie. A oznacza to przecież, iż uznali oni, że synteza jądrowa w temperaturze pokojowej jest możliwa i to zupełnie inną drogą niż się tego wcześniej spodziewano!

Poza wszystkimi innymi wnioskami — jest to w moim rozumieniu bardzo pozytywny fakt. Fakt w gruncie rzeczy romantyczny, gdyż pokazujący, że można zrobić coś dużego w nauce bez potrzeby korzystania z urządzeń, na które stać tylko nieliczne kraje i do których dostęp, siłą rzeczy, mogą mieć tylko nieliczni naukowcy. Zbudowanie „tokamaka" przewyższa już dzisiaj nawet możliwości krajów najbogatszych (dlatego właśnie podjęto międzynarodowy program badań nad kontrolowaną syntezą termojądrową). Natomiast na szklaną probówkę, elektrodę z tytanu i pół litra „ciężkiej wody" stać już wiele ośrodków naukowych, a nawet indywidualnych uczonych.

I mimo, że piszę o syntezie jądrowej, to nic nie stoi przecież na przeszkodzie, aby te ogólne wnioski, mówiące, że w nauce możliwe są wielkie niespodzianki, których autorami może być każdy z nas (jeśli będzie oczywiście bardzo chciał), Czytelnicy „Bajtka" przetransportowali na swoje własne podwórko.

Waldemar Siwiński

ZA MIESIĄC:

- rozmowa z komputerem po polsku
- ankieta „Bajtka" — czyli co o nas myślą Czytelnicy
- język „C" dla najmłodszych
- kolejne kluby w „Złotej Dyskietce Bajtka"
- nowe gry
- prawo Archimedesa inaczej
- stałe rubryki
- technologia mikroprocesorów

Bajtek

„BAJTEK" — MIESIĘCZNY DODATEK DO „SZTANDARU MŁODYCH"

ADRES: 00-687 Warszawa, ul. Wspólna 61. Tel. 21-12-05 Przewodniczący Rady Redakcyjnej: Jerzy Domański — redaktor naczelny „Sztandaru Młodych".

ZESPÓŁ REDAKCYJNY: Waldemar Siwiński (z-ca redaktora naczelnego „SM" — kierownik zespołu „Bajtka"), Grzegorz Onichimowski (sekretarz redakcji „Bajtka"), Roman Poznański (kierownik działu klanów), Krzysztof Czernek, Sławomir Gajda (red. techniczny), Andrzej Pilaszek, Sławomir Polak, Wanda Roszkowska (opr. graficzne), Kazimierz Treger, Marcin Waligórski, Roman Wojciechowski. Zdjęcia w numerze: Leopold Dzikowski

klany redagują:
Commodore — Klaudiusz Dybowski, Dominik Falkowski
Amstrad-Schneider — Jonasz Mayer
Spectrum — Marcin Przasnyski
Atari — Wojciech Zientara, Sergiusz Piotrowski

Fotoskład — Tadeusz Olczak,
Montaż offsetowy — Grażyna Ostaszewska,
Korekta — Maria Krajewska, Zofia Wóltańska

WYDAWCA: RSW „Prasa-Książka-Ruch" Młodzieżowa Agencja Wydawnicza, al. Stanów Zjednoczonych 53, 04-028 Warszawa. Telefony: Centrala 13-20-40 do 49, Redakcja Reklamy 13-20-40 do 49 w. 403, 414.

Cena 150 zł.

Skład techniką CRT-200, przygotowalnia offsetowa i druk: PRASOWE ZAKŁADY GRAFICZNE RSW „PRASA-KSIĄŻKA-RUCH" w Ciechanowie, ul. Sienkiewicza 51.

Nr zlecenia 033259 n. 120000
A-111





Kolorowa kserokopiarka firmy Brother

Na tegorocznych targach wszechwładnie królowała telekomunikacja

CO TAM, PANIE, W KOMPUTERACH?

Święto komputerów trwa w Europie tydzień w ciągu roku — na wiosnę. Wtedy właśnie wszyscy, którzy mają cokolwiek do powiedzenia w tej jednej z największych gałęzi przemysłu współczesnego świata spotykają się w jednym miejscu — na terenach targowych Hanoweru. A spotkanie, o którym mowa, zwie się CeBIT.

Targi CeBIT 89 nie odbiegały od normy — były olbrzymie. W halach o łącznej powierzchni ponad 230 tys. metrów kwadratowych swoją ofertę zaprezentowało 3125 wystawców z 37 krajów. Obecni byli giganci — IBM, Apple, Commodore, Toshiba, czy Microsoft, ale także setki małych lecz niezwykle dynamicznie się rozwijających firm z Tajwanu, Singapuru, czy Korei Południowej. Ilość prezentowanych urządzeń, programów, sieci itd. była tak zatem tak wielka, że chociaż niniejsza relacja nie będzie z pewnością pierwszym sprawozdaniem z CeBIT-u jakie zobaczycie w naszej prasie (wśród setek dziennikarzy akredytowanych przy targach znalazła się i spora polska gromadka) jednak mam pewność, że nie powtórzę wiele za kolegami po piórze. Na postawione w tytule pytanie każdy znalazł pewnie swoją odpowiedź.

Zanim przejdę do swojej chciałbym... skorygować nieco pytanie. CeBIT, chociaż jego hale po dach niemal zapełnione są „komputerowym złomem" nie jest bynajmniej targami komputerowymi. Jego obserwacja raz jeszcze udowadnia nam tezę, jaką nie raz stawialiśmy już przed naszymi czytelnikami. Komputery same w sobie, niezależnie od pojemności pamięci, szybkości działania itd. mało kogo już podniecają. O wiele bardziej istotne są zadania jakie muszą one spełniać. Do nich dostosowuje się konstrukcję i parametry urządzeń-komputerów, peryferiali, sieci, a także softwaru.

Na tegorocznych targach wszechwładnie królowała telekomunikacja. Jej rozwój determinuje dziś w dużym stopniu dalszy postęp cywilizacyjny. Cóż bowiem z tego, że zdołamy przy pomocy komputera przetworzyć otrzymaną informację bądź odtworzyć ją z pamięci, jeśli informacji tej nie będziemy potrafili szybko i tanio przekazać dalej, a samemu sięgnąć po inną, niedostępną nam jeszcze pod ręką?

Wystawiające na hanowerskich targach firmy miały na to oczywiście swoje recepty. Najczęstsza z nich brzmiała: ISDN (integrated services digital network). Opisywana tym skrótem sieć połączeń umożliwia przekazywanie przy pomocy tego samego kabla przekazywanie dźwięku słów, obrazu TV, oraz oczywiście danych komputerowych. Nie jest ona przy tym żadną ciekawostką, czy prototypem. Zajmujący na wystawie niemal całe piętro jednej z hal Federalny Urząd Poczty pokazał, dla przykładu, działającą już w dużej części RFN sieć bliską zintegrowanej. Jak stwierdzono na seminarium, które odbywało się równolegle do targów, w 1993 roku sieć ta pokryje całe terytorium nadreńskiej republiki.

W telekomunikację ostatnich lat przebojem wdarły się telefaksy. Dziś mówi się już o „kulturze faxów". Producenci tych urządzeń mają z pewnością powody do zadowolenia, natomiast w przeddzień plajty znajdują się już chyba wszyscy ci, którzy postawili na teleksy. Pozostaje im już chyba wyłącznie eksport do Polski.

Sieci telekomunikacyjne, telefony, telefaksy. Co mają one do komputerów? Okazuje się, że dużo i coraz więcej. CeBIT pokazał, jak wielu producentów zarówno sprzętu jak i oprogramowania interesuje się dziś komunikacją pomiędzy publicznymi sieciami telekomunikacyjnymi, a ich lokalnymi odpowiednikami, sieciami komputerowymi przedsiębiorstw, banków.

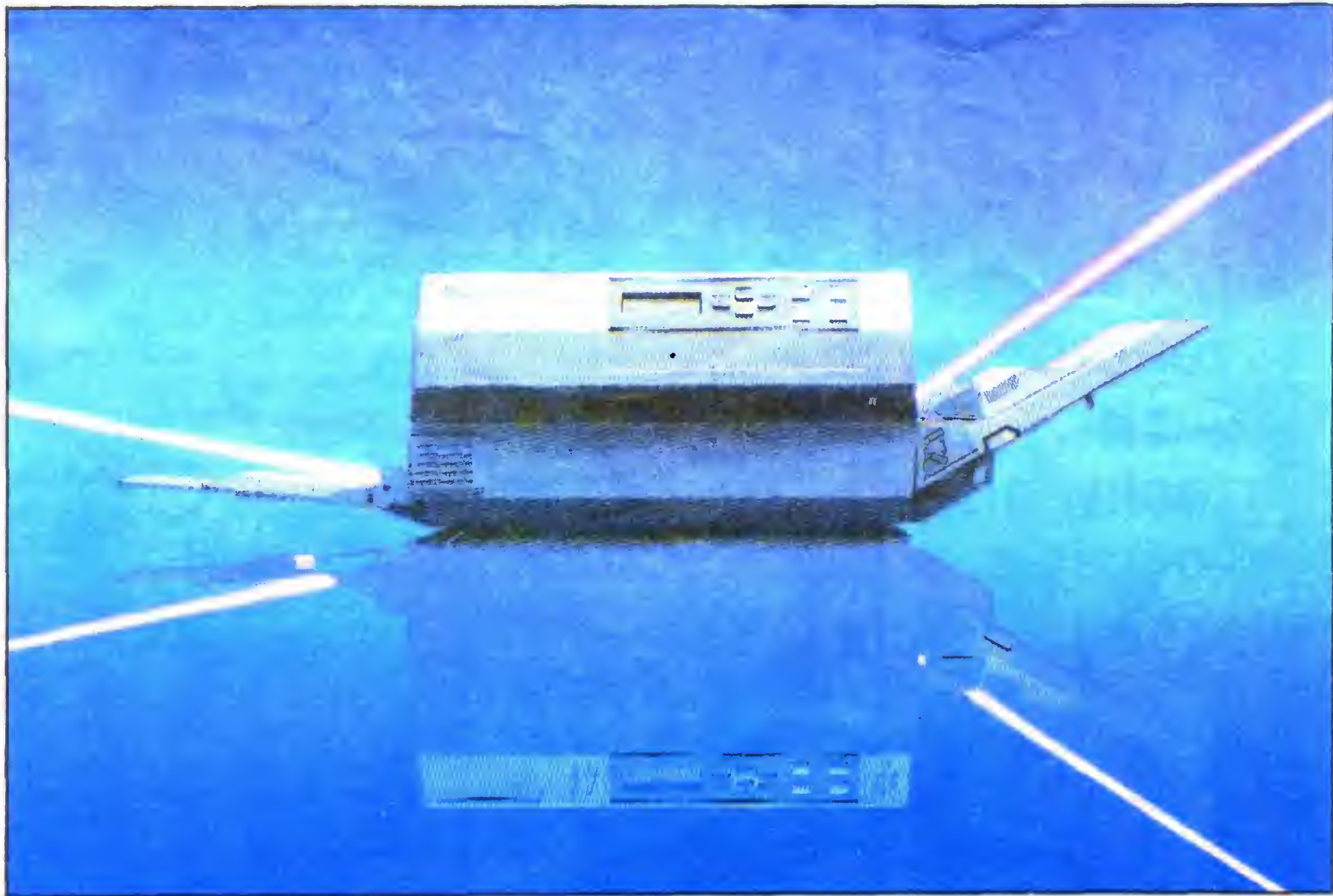
Pora jednak od tych (bez względu na ambitne plany przedsiębiorstwa Polska Poczta Telegraf i Telefon), egzotycznie brzmiących, wieści przejść do tego, co bliższe naszym czytelnikom, do rynku komputerów domowych i osobistych. Te pierwsze zaczynają już powoli odchodzić od historii. Z naszych „klanowych" maszyn na scenie pozostał już tylko Commodore 64 prezentowany jednak raczej jako ciekawostka. Atari 520 ST i Commodore Amiga 500, czyli komputery domowe końca lat osiemdziesiątych, natomiast, chociaż dziś bardzo popularne, cenowo nie odbiegają od najprostszych PC, takich jak choćby Commodore PC 10 lub Schneider Euro PC. Te ostatnie mają natomiast te przewagę nad konkurentami, że ich bardziej rozbudowane odpowiedniki znajdują się praktycznie w każdym biurze i w każdej szkole. (oczywiście na Zachodzie). Zatem dysponując PC w domu zawsze możemy korzystać wymiennie z komputerów przed i po fajerancie.

Umacnianie się monopolu komputerów PC opartych na mikroprocesorach linii Intela (8088, 80286, 80386) nie dokonuje się bynajmniej poprzez lawinę wynalazków technicznych. W zasadzie żaden z potenta-

tów rynku hardwarowego nie pokazał w Hanowerze niczego rewelacyjnego. Jedynie dalekowschodni producenci próbowali koniecznie zaszokować zblazowanych Europejczyków szybkością działania swych komputerów. O ile w poprzednich latach na ekranach monitorów królowały niepodzielnie sympatyczne małpki i dziewczęta w przezroczystych bluzeczkach tym razem na tajwańskich i singapurskich stoiskach obejrzeć było można obrazki znacznie mniej interesujące, przynajmniej dla niewtajemniczonych. Prawie wszystkie komputery na okrągło pracowały z programem testującym szybkość, a na ekranach migały coraz bardziej szokujące liczby — 25 MHz, 30 MHz, 35 MHz, itd. Żadna jednak z tych liczb nie wzruszała zbytnio tłumnie zwiedzających targi biznesmanów. Stanowczo woleli oni tłoczyć się tam, gdzie prezentowano nie najszybsze nawet, pojedyncze komputery, lecz przy gotowych już systemach bankowych, komputerowego wspomagania produkcji, czy projektowania.

W dalszym ciągu zawrotną karierę robią na rynku przenośne walizkowe PC, najczęściej kompatybilne już z AT, wyposażone w plazmowe ekrany (także kolorowe!), ważące od 3 do 6 kilogramów i mieszczące się swobodnie w teczce. Rekord na polu miniaturyzacji pobiło tym razem Atari dzięki modelowi PC Folio- kieszonkowemu komputerowi w pełni kompatybilnemu z PC XT. Ta zadziwiająca maszynka, której pełną charakterystykę zamieścimy w „Bajtku" „Tylko o Atari 3" nie posiada oczywiście stacji dysków ani dysku twardego (jest na to po prostu za mała), za to można się nią posługiwać nie tylko w podróży, ale nawet na przyjęciu bez obawy zbytniego wypchania kieszeni fraka.

Ponieważ polscy dziennikarze na przyjęciach bywają rzadko, a chcieliby jednak dysponować przenośną maszyną do pisania z kalendarzem, szukaliśmy wśród portabli czegoś większego od PC Folio z ceną nie przekraczającą granic wyobraźni. Nie było to łatwe, kupując taki komputerek trzeba liczyć się z wydatkiem rządu 1500–2000 dolarów (bez dysku twardego). Na dodatek zdecydowany lider w produkcji „maluchów" japońska Toshiba oznajmiła nam poprzez swoich przedstawicieli, że ze Wschodem, niestety, już nie handluje. W tej sytuacji najbardziej zainteresował nas, przedstawiany już na łamach „Bajtka", Cambridge Z88, ostatnie dziecko lorda Clive'a Sinclaira. W cenie ok. 400 dolarów komputerek ten jest naprawdę ciekawą propozycją dla dziennikarza, czy ruchliwego biznesmana.



Drukarka laserowa GO-5000

Jak już wspomniałem, wielu prawdziwie nowatorskich komputerów na CeBIT-cie nie prezentowano. Inaczej natomiast rzecz się miała chociażby z oprogramowaniem i urządzeniami peryferyjnymi. Najbardziej wśród producentów drukarek zaskoczył nas „Star". Firma ta, praktyczny monopolista na polskim rynku drukarek wystartowała z dwiema nowymi seriami swoich produktów. Pierwsza z nich, oznaczona symbolem FR, obejmuje drukarki typu „heavy duty" szybkie, trwałe o dużej pojemności bufora, natomiast druga, XB zastąpić ma, jak wróble ćwierkają, tak popularne NX i NB. Nowe „Stary" wyróżniają się elegancją i prostotą. Chyba przyjmą się i u nas, jako że firma obiecuje wprowadzenie na stałe do ich „menu" także i polskiego alfabetu.

Jeśli już mowa o Polsce warto wspomnieć o rodzimych akcentach CeBIT-u. Na początek ten bardziej optymistyczny. Wśród setek mniejszych firm, prezentujących się na targach znalazła się jedna polska — a właściwie działająca w Polsce-PZ „Emix". Zaprezentował on dwa pomysłowe systemy służące wspomaganiu pracy lekarzy — system intensywnego nadzoru medycznego przetwarzający w PC analogowe sygnały biegnące od urządzeń diagnostycznych i weryfikujący niejako na bieżąco pełny obraz stanu zdrowia pacjenta oraz system ewidencji chorych. Mimo skromnych rozmiarów ekspozycji „Emixu" i braku odpowiedniej jakości materiałów reklamowych, jak zapewnił jego dyrektor Kazimierz Tuzimski, firma nic nie straciła na udziale w targach. Przeciwnie, oba systemy, a także prezentowane również na stoisku komputery i zasilacze do nich znalazły sporo potencjalnych nabywców. Z kilkoma szpitalami w RFN kontrakty już podpisano.

Znacznie mniej budujący widok przedstawiało „oficjalne" polskie stoisko firmowane przez szacowne PHZ „Metronex". Stały na nim antyczne niemal maszyny do pisania i drukarki D 100 z Błonia, niewątpliwie tanie i trwałe, lecz wyglądem, w konfrontacji ze „Starami", „Epsonami", czy „Oki" nasuwające porównanie między „Syreną" a Fiatem Tipo. W tym stanie rzeczy, jak stwierdziliśmy zgodnie z red. Majewskim z „Komputera" nalegać będziemy, by w przyszłym roku zamiast „Metronexu" wysłać na CeBIT „Cepelię". Skoro i tak wiadomo, że reprezentujemy skansen...

Ale żarty na bok. Jak trzeba wchodzić na rynki obce pokazały nam w Hanowerze Indie. „Business with India", dzięki zapobiegliwości indyjskich instytucji rządowych stał się przewodnim tematem tegorocznego CeBIT-u. Chociaż prezentowane na wielkim narodowym stoisku Indii urządzenia i programy nie reprezentowały wcale olśniewającego poziomu, sporo z nich zapewne sprzedano, zaś inne zainspirowały zachodnich biznesmanów do utworzenia wspólnych przedsięwzięć. Jak zapewniał nas jeden z dyrektorów targów, całkiem możliwe byłoby zorganizowanie za 2 lata podobnej akcji „Business with Poland". Aby jednak mieć co pokazać nasze Ministerstwo Współpracy Gospodarczej z Zagranicą i centrale handlu zagranicznego muszą szerzej sięgnąć po wyroby małych firm i spółek, także prywatnych, zapewnić im promocję, zapłacić nawet za stoiska i materiały reklamowe dla tych, którzy jeszcze nie zarabiają sami dolarów, lecz ich produkcja daje im takie potencjalne szanse. To, że ewentualne zyski wpłynęłyby do prywatnych, głównie, kieszeni nikogo nie powinno martwić. Wszak najważniejsze jest, że pieniądze te znajdą się w Polsce, a ich właściciele będą płacić podatki.

Po tej przydługiej nieco uwadze „CeBIT" a sprawa polska" wróćmy, na koniec, znów do techniki. Otóż, jako się już rzekło tegoroczne targi nie ukazały żadnych rewolucyjnych zmian w technice i technologii wytwarzania komputerów. Zauważyć było można nawet odwrotne zjawisko. Pokazywane przed rokiem, czy dwoma laty wynalazki, które przynieść miały przewrót w technice bardzo wolno wchodzą na szeroki rynek. W miarę jak rośnie ilość używanych w poszczególnych branżach komputerów rozszerza się, zasięg sieci komputerowych, ich wzajemna komunikacja itd. coraz trudniej jest zmieniać standard. I tak nie nastąpił, bynajmniej jeszcze, zmierzch stacji dyskietek 51/4 cala. Nowy standard -31/2" obowiązuje, na razie tylko w światku komputerów przenośnych. W dużych PC używa się często obu najpopularniejszych formatów. Znacznie mniej niż przed rokiem promocy wano w Hanowerze pamięci CD-ROM, czyli optycznych dysków wielkiej pojemności. Wszystko wskazuje na to, że upowszechnią się one tylko wówczas, gdy najważniejsi producenci zdecydują się między sobą na uzgodnienie jakiegoś standardu i namówią wytwórców komputerów, by ze standardu tego skorzystali.

Ten swoisty wymuszony konserwatyzm, mniej widoczny np. w branży drukarek laserowych, czy kserokopiarek, gdzie pokazano znowu szereg „cudeniek" jest trochę pozorny. Po prostu dziś myśli się głównie o upowszechnienie i praktycznym zastosowaniu tego, co już wymyślono. Cywilizacja goni jakby za ostatnimi swoimi wytworami. Na nas, oczywiście, nikt nie zaczeka. Tegoroczny CeBIT, wskazuje jednak, że technika rozwijająca się bardziej „wszerz" aniżeli „w przód" może być zagrożeniem, ale i szansą.



Maszyna do liczenia banknotów

Grzegorz Onichimowski

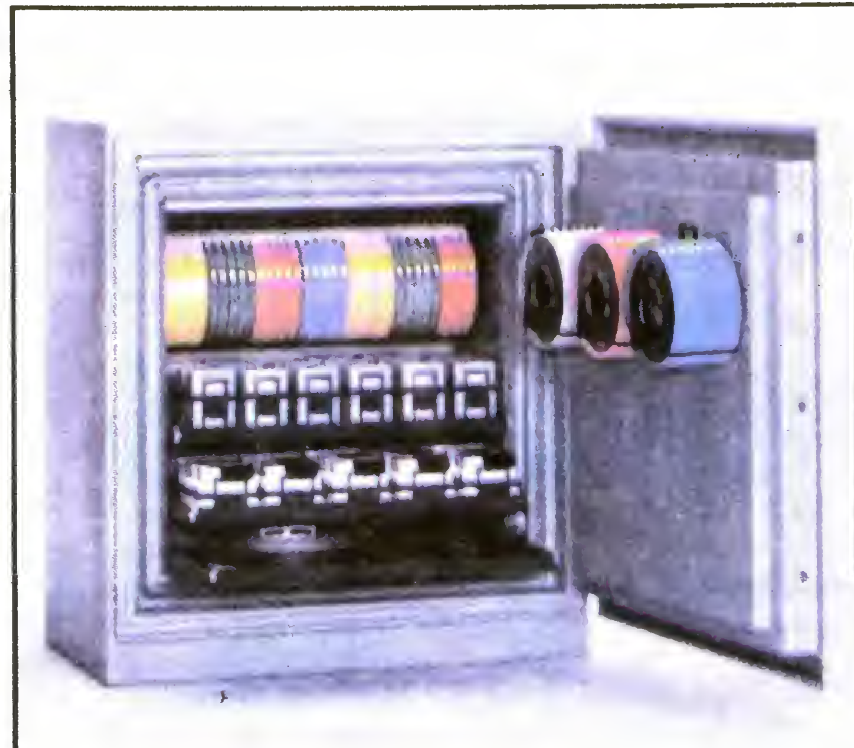


KOLOR Z KOMPUTERA

Dzięki coraz to doskonalszym kartom graficznym i monitorom świat na ekranach mikrokomputerowych jest coraz bardziej kolorowy. Jak dotychczas, jednak, nie bardzo radzono sobie z uwiecznianiem go na papierze. Prezentowane powyżej urządzenie — kolorowa wideodrukarka SV 6500 firmy Kodak częściowo rozwiązuje ten problem. Ta termiczna drukarka przyjmować może do uwieczniania obraz zarówno z kamery systemu U-matic (profesjonalnej), NTSC (amatorskiej), z gramowidu (magnetowidu) a także, poprzez klasyczne złącze Centronics z komputera klasy PC. Wprawdzie obrazki wychodzące z tego urządzenia mają tylko dwa formaty (w zależności od tego, czy pochodzą z wejścia wideo, czy z Centronicsa) lecz za to śmiało konkurować mogą pod względem jakości ze średniego standardu zdjęciami. Paleta kolorów — 256 zadowolić może nawet wybrednych użytkowników. Kaseta ze specjalną taśmą drukującą wystarcza na zrobienie ok. 100 odbitek.

SV 6500 jest tylko jednym z elementów systemu przetwarzania obrazów proponowanego przez Kodak. W jego skład wchodzą ponadto: projektor do przetwarzania na obraz cyfrowy konwencjonalnych zdjęć, stacja wideodysków zapisująca po 50 pełnoekranowych obrazów kolorowych na jednej specjalnej 2-calowej dyskietce, a także magnetowidy, kamery i, rzecz jasna mikrokomputery. Systemy takie, nieocenione zwłaszcza w edukacji są jednak, przynajmniej do dziś stanowczo za drogie na kieszeń naszych szkół.

— FORT KNOX — Z DYSKIETKAMI



Problem bezpiecznego przechowywania danych, nierzadko bezcennych nie należy wcale do łatwych do rozwiązania.

Czasem nie wystarcza tylko sporządzanie kopii bezpieczeństwa, trzeba także zabezpieczyć same dyskietki i to zarówno przed niepowołanymi użytkownikami jak i np. przed pożarem. Do tego celu niemal powszechnie używa się dziś najrozmaitszych sejfów. Prezentowany na fotografii jest dziełem jednego z zakładów zachodnioniemieckiego koncernu Nixdorf. Mieści w swoim wnętrzu 2160 dyskietek 51/4" lub 2000 dyskietek 31/2" lub 34 taśmy magnetyczne 10". Waga tego urządzenia — 543 kg, natomiast cena 4.500 marek zachodnioniemieckich. Dla przezornych z kraju, gdzie nie kopiuje się bezkarnie programów szafa ta warta jest z pewnością swojej ceny

Miasto przyszłości

Jeszcze przed rokiem 2000 w miejscowości Itichara, niedaleko od Tokio (prefektura Tiba), pojawi się pierwsze w Japonii „mini-miasto" przyszłości, które ma stać się wzorcem do wznoszenia metropolii XXI wieku.

U podstaw projektu znajduje się idea o połączeniu za pomocą komputerów w jedną całość domów mieszkalnych, budynków administracyjnych, transportu i wszystkich służb miejskich. Ułatwić to powinno mieszkańcom wykonywanie wszystkich codziennych ich funkcji i prac, oraz usprawnić zarządzanie całością. Do realizacji projektu włączyło się już wiele znanych firm japońskich, m.in. takie giganty, jak „Sony" i „Casio". (ws)

Skomputeryzowani chałupnicy

Gdy jeszcze niewdano Alvin Toffler przedstawiał w „Trzeciej fali" wizję nowego biura, a właściwie nowego przedsiębiorstwa, polegającego na tym, że... go nie ma, to znaczy pracownicy nie muszą zbierać się codziennie rano w jednym budynku, aby móc zbiorowo funkcjonować — wydawało się to może już wprawdzie nie fantazją, ale jednak przyszłością bardzo odległą.

I oto okazuje się, że takie biura już nie tylko istnieją, ale również implikują problemy, którymi muszą zajmować się najwyższe gremia państwowe. Tak właśnie jest w Japonii, gdzie tamtejsze Ministerstwo Pracy przedłożyło parlamentowi dokument, wzywający do przyjęcia ustawy, mającej na celu ochronę praw „skomputeryzowanych chałupników".

Głównymi czynnikami technicznymi, które umożliwiły pojawienie się skomputeryzowanych miejsc pracy w domach, były: rozwój sieci telekomunikacyjnej oraz miniaturyzacja komputerów osobistych. Natomiast od strony społeczno-gospodarczej upowszechnienie się takiego rozwiązania zostało spowodowane dążeniem wielu firm do zwiększenia swej wydajności pracy, aby poprawić sobie pozycję w warunkach walki konkurencyjnej.

Firmy instalują bezpłatnie w domach chętnych praktycznie całą niezbędną aparaturę Domowe terminale podłączone są do centralnego komputera, znajdującego się w głównej siedzibie firmy, niekiedy w odległości wielu kilometrów od mieszkania pracownika. Dzięki temu urzędnicy, nie wychodząc z domu, mogą dokonywać różnych obliczeń, zajmować się projektowaniem, programowaniem, pracami redaktorskimi (opracowywaniem tekstów, „łamaniem" kolumn itp.), prowadzeniem buchalterii i wieloma innymi pracami.

Taki rodzaj pracy jest atrakcyjny przede wszystkim dlatego, że całkowicie rozwiązuje problem transportu i umożliwia samodzielne regulowanie harmonogramu dnia. Nic więc dziwnego, że ponad 90 proc. skomputeryzowanych chałupników to właśnie kobiety, mające wyższe wykształcenie. (ws)



NOWE DZIECI ATARI

Atari rozszerza paletę swoich wyrobów. Gdy modele 520/1040 ST ostatecznie przeszły do kategorii komputerów domowych zdobywając nawet nagrodę CHIP-a dla najlepszych komputerów domowych 1988 roku firma postawiła w rozwoju swych systemów profesjonalnych na komputery zgodne ze standardem DOS. Najnowsze efekty tej nowej strategii to maleńki Atari PC Folio (piszemy o nim obok i w „Tylko o Atari 3") oraz prezentowany na zdjęciu model PC 4.

Komputer ten o dość „staromodnym" wyglądzie zewnętrznym kryje w sobie szereg nowoczesnych, niestandardowych rozwiązań technicznych. W pełni kompatybilny z PC AT wyposażony jest w procesor 80286 o częstotliwości zegara 8 i 12 MHz. Do standardowego wyposażenia tej maszyny należy nowoczesna ergonomiczna klawiatura „z potwierdzeniem" oraz typowa mysz Atari.

Ciekawie rozwiązano w tym modelu problem pamięci. Obok RAM umieszczonego na płycie głównej (1–8 MB) zainstalowano stację dyskietek 31/2" oraz wymienny dysk szybkiego dostępu 44 MB. Opcjonalnie towarzyszy im także „konwencjonalny" dysk twardy o pojemności 60 MB

Czy ujrzymy kiedyś tę maszynę na półkach naszego „Peweksu" obok 65 XE i 520 ST?

Pierwsze kroki w asemblerze

MONITORY (2)

W dalszym ciągu naszego wprowadzenia do programowania w języku maszynowym zostaną przedstawione dwa kolejne monitory: Happy Mon i UMon XL.

HAPPY MON

Uruchamianie **Happy Mon** przebiega odmiennie od innych programów. Najpierw trzeba wczytać program, który będzie badany, ale bez uruchamiania go. Uzyskuje się to przy odczycie z DOS-u przez napisanie na końcu nazwy pliku „/N". Potem dopiero wczytuje się **Happy Mon**. Jeśli jest to wersja, która nie uruchamia się sama, to robimy to funkcją „M" DOS-u podając adres 8000.

Monitor rozpoznaje liczby dziesiętne i szesnastkowe. Ich zapis jest zgodny z ogólnie przyjętą konwencją, czyli liczby szesnastkowe muszą się rozpoczynać od znaku dolara „$", zaś liczby dziesiętne są pisane normalnie.

Happy Mon nie posiada normalnych instrukcji zapisu i odczytu. Badany program musi być odczytany przed uruchomieniem monitora, a jego zapis jest możliwy po przejściu do DOS-u lub Basica. Istnieją jednak dwa polecenia umożliwiające odczyt i zapis pojedynczych sektorów. Pierwsza operacja jest wykonywana po poleceniu **R** z numerem sektora i adresem bufora. Analogicznie, zapis jest dokonywany po poleceniu **W**.

Polecenie **O** z nazwą pliku pozwala na zapisywanie rezultatów pracy monitora w drugim pliku (oprócz ekranu). Może to być plik dyskowy lub drukarka (0 #P:). Zamknięcie tego pliku następuje po poleceniu **N** (bez żadnych parametrów).

Bogaty jest zestaw poleceń służących do wykonywania operacji na pamięci komputera. Przeglądanie zawartości pamięci jest wykonywane po podaniu **Z** lub **L**. Pierwsze polecenie wyświetla tylko wartości szesnastkowe, a drugie także znaki ATASCII. Po wyświetleniu ośmiu bajtów pojawia się pytanie „Continue (Y/N)". Naciśnięcie N kończy przeglądanie pamięci, a Y lub RETURN wyświetla następne osiem bajtów. Pytania tego można uniknąć wpisując na końcu polecenia literę **N**.

Pojedyncze bajty wpisuje się do pamięci przy użyciu polecenia **P**, które działa podobnie do POKE w Basicu. Słowa dwubajtowe (o wartości od 0 do 65535) są natomiast wpisywane poleceniem **V**. Do wpisania większej liczby bajtów należy użyć polecenia **A** z adresem. Wyświetlany jest wtedy adres i można wprowadzić do ośmiu wartości. Po naciśnięciu RETURN podane bajty są umieszczane w pamięci i pojawia się następny adres. Wpisywanie przerywa się klawiszem RETURN. Bardzo podobnie działa polecenie **E**, lecz dodatkowo wyświetla ono dotychczasową zawartość pamięci. Zmian można także dokonywać po poleceniach **L** i **Z**, jeśli wprowadzi się kursor do wyświetlanego wiersza. UWAGA: W poleceniach **A**, **E**, **L** i **Z** można podawać tylko liczby szesnastkowe bez znaku dolara. Gdy zachodzi potrzeba wpisania tekstu (znakami ASCII), to można użyć polecenia **T**. Dodatkowe udogodnienie daje polecenie **F**, które służy do wypełnienia bloku pamięci jedną wartością.

Bloki pamięci można jeszcze przenosić i przeszukiwać. Pierwsza z tych czynności jest wykonywana po poleceniu **M**, które musi zawierać adres początku bloku, nowy adres i liczbę przepisywanych bajtów. Do przeszukiwania pamięci służą polecenia **HB** i **HW**, które pozwalają odnaleźć odpowiednio bajt lub słowo. Po znalezieniu wyświetlany jest napis „Continue (Y/N)". Podobnie, jak było to wcześniej opisane, na końcu poleceń **HB** i **HW** można umieścić literę **N**.

W każdym monitorze bardzo ważna jest możliwość disasemblacji (zmiany kodu liczbowego na mnemoniki asemblera). **Happy Mon** dysponuje w tym celu poleceniem **D**, które wyświetla mnemoniki razem z ich adresami. Dodatkowe polecenie **U** wyświetla mnemoniki z numerami wierszy zamiast adresów. Taki rezultat można wykorzystać bezpośrednio w dowolnym asemblerze (oczywiście po zapisaniu na dyskietce przy pomocy 0).

Ubogie są możliwości śledzenia programu znajdującego się w pamięci. **Happy Mon** ma tylko polecenie **G**, które powoduje skok do podanego adresu i wykonywanie znajdującego się tam programu. Jeśli program ten nie kończy się rozkazem RTS, to powrót do monitora jest niemożliwy.

Ponadto istnieje kilka poleceń spełniających pomocnicze funkcje w monitorze. Polecenie **?** z liczbą dziesiętną powoduje wyświetlenie wartości tej liczby w systemie szesnastkowym i odwrotnie. Polecenie * ustala wartość licznika programu użytkownika. Działa ono tak samo, jak * = w MAC/65. Jeżeli nie odpowiada nam barwa ekranu, to polecenie **K** zamienia kolory tła i znaków.

Zakończenie pracy **Happy Mon** umożliwiają polecenia **X** i **Q**. Pierwsze z nich powoduje powrót do DOS-u, a drugie do Basica (jeżeli przy uruchomieniu nie był wciśnięty klawisz OPTION).

UMon XL

UMon XL jest monitorem działającym w nieco inny sposób od opisanych poprzednio. Wszystkie polecenia są w nim jednoliterowe i powodują wyświetlenie pytań określających parametry wykonania polecenia. Dozwolone są wyłącznie liczby szesnastkowe. Ponieważ naciśnięcie klawisza **HELP** wyświetla wykaz poleceń na ekranie, to ich szczegółowy opis można pominąć. Ograniczamy się tylko do wyjaśnienia zawartych tam określeń.

A — Wyświetlanie katalogu dyskietki znajdującej się w stacji dysków o numerze podanym na pytanie „DRIVE".

B — Przejście do Basica. Jeśli Basic jest odłączony, to komputer się zawiesza.

DODATKOWE KLAWISZE INACZEJ

W „Bajtku" 1/89 opisany był sposób zainstalowania w komputerze czterech dodatkowych klawiszy. Spełniają one funkcje klawiszy F1—F4 znajdujących się w modelu 1200 XL. Uzyskanie takiego efektu nie wymaga jednak żadnych przeróbek komputera.

Działanie klawiszy Atari jest określone przez znajdującą się w pamięci ROM tablicę definicji klawiszy (*Key Definitions Table*). System operacyjny korzysta z tej tablicy poprzez wektor KEDEFP, który jest umieszczony w komórkach 121 i 122 pamięci RAM. Każdy Czytelnik domyśla się już, że wystarczy napisać nową tablicę, umieścić ją gdziekolwiek w pamięci komputera i zmienić wektor KEDEFP.

Niestety, jest to zbyt proste, aby było prawdziwe. Klawisze F1, F2 i F4 naciśnięte razem z CONTROL są obsługiwane bezpośrednio przez procedurę przerwania wywołaną naciśnięciem klawisza. Tylko pozostałe można więc zmienić w podany wyżej sposób. Ponieważ zmiana procedury przerwania wymaga nieco więcej miejsca w pamięci, to zajmiemy się nią później.

Klawisze funkcyjne naciśnięte pojedynczo przesuwają kursor. Służą do tego również klawisze ze strzałkami naciskane razem z CONTROL. Pozostaje więc tylko pięć kombinacji: F1—F4 z SHIFT oraz F3 z **CONTROL**. Dla wykonywanych przez nie funkcji należy wybrać nowe klawisze. Oczywiście wybrać można tylko takie, które nie mają żadnej funkcji albo ich funkcje są dublowane przez inne klawisze. Zrobiłem to następująco:

SHIFT+CLEAR
kursor do lewego górnego rogu
SHIFT+INVERSE
kursor do lewego dolnego rogu
SHIFT+ESC
kursor do lewego marginesu
SHIFT+RETURN
kursor do prawego marginesu
CONTROL+ESC
włącza i wyłącza dźwięk klawiatury

Teraz czas na program. Znajdujemy tablicę klawiszy w ROM i jej adres przypisujemy zmiennej TK (wiersz 50). Następnie przepisujemy całą tablicę (ma ona 192 bajty) na nowe, bezpieczne miejsce — od adresu 1536, czyli na szóstą stronę pamięci (wiersz 60). Zmiany w tablicy wykonywane są przez drugą pętlę (wiersz 70). Odczytuje ona kolejno z instrukcji DATA w wierszu 90 pary liczb. Pierwsza liczba oznacza kod kombinacji klawiszy i stanowi adres (liczony od początku tablicy), pod którym zapisywana jest druga z liczb określająca funkcję wykonywaną przez te klawisze. Na zakończenie ustalamy nową wartość wektora KEYDEFP (wiersz 80).

Pokazany program może być po uruchomieniu usunięty z pamięci. Szósta strona pamięci jest bardzo bezpiecznym miejscem i zawartej tam tablicy nie niszczy ani naciśnięcie RESET, ani wczytanie innego programu. Po RESET trzeba tylko powtórzyć instrukcje POKE umieszczone w wierszu 80.

Wojciech Zientara

```
YL   10 REM KLAWISZE FUNKCYJNE
FN   20 REM Wojciech Zientara
AK   30 REM Copyright (c) Bajtek
BB   40 REM
KU   50 TK=PEEK(121)+256*PEEK(122)
GP   60 FOR I=0 TO 191:POKE 1536+I,PEEK(TK+
        I):NEXT I
TM   70 FOR I=0 TO 4:READ A,B:POKE 1536+A,B
        :NEXT I
IJ   80 POKE 121,0:POKE 122,6:END
NH   90 DATA 118,138,103,139,92,140,76,141,
        156,137
```

C — Zmiana zawartości pamięci. Wymaga podania adresu (ADDRESS), następnie wyświetlana jest dotychczasowa zawartość wskazanej komórki. Samo naciśnięcie RETURN kończy funkcję.

D — Przeglądanie zawartości pamięci od adresu podanego na pytanie „FROM". RETURN powoduje kontynuację, a każdy inny klawisz przerwanie realizacji funkcji.

E — Redagowanie sektora dyskietki. Wymaga podania numeru stacji i numeru sektora. Odczytany sektor można wypełnić zerami (CLEAR SECTOR Y/N). Zmiana zawartości sektora jest wykonywana liczbami lub znakami (HEX OR ASCII EDIT). Oczywiście sektor można potem zapisać z powrotem na dyskietkę (WRITE SECTOR).

F — Wypełnienie bloku pamięci od „FROM" do „TO" wartością „WITH".

G — Przejście do wykonywania programu użytkownika od podanego adresu.

H — Wyświetlenie zawartości sektora dyskietki o podanym numerze. Klawisze ze strzałkami powodują zmniejszenie lub zwiększenie numeru sektora.

I — Formatowanie dyskietki we wskazanej stacji dysków.

J — Wykonanie procedury w języku maszynowym od podanego adresu. Kontynuacja i zakończenie jak w poleceniu **D**.

M — Przemieszczenie bloku pamięci zaczynającego się od sektoru „FROM" i kończącego się na „TO". Blok jest zapisywany od adresu „NEW".

P — Drukowanie na drukarce całej zawartości ekranu.

R — Odczyt ze stacji dysków „DRIVE" sektorów o numerach od „FROM" do „TO". Odczytane dane są umieszczone w pamięci od adresu „RAM BUFFER".

S — Przeszukiwanie zawartości pamięci (MEMORY) lub dyskietki (DISK). Poszukiwany ciąg można podać w postaci liczb szesnastkowych (HEX) lub znaków kodu ASCII.

V — Porównanie zawartości obszaru pamięci od „FROM" do „TO" z obszarem rozpoczynającym się od „WITH".

W — Zapis sektorów na dyskietce (podobnie do **R**).

— Zmiana liczby szesnastkowej na dziesiętną

$ — Zmiana liczby dziesiętnej na szesnastkową

% — Zmiana liczby szesnastkowej na dwójkową.

& — Zamiana liczby dwójkowej na szesnastkową

Wojciech Zientara

Składnia poleceń Happy Mon

A adr_pocz
D adr_pocz [.N]
E adr_pocz
F adr_pocz, dł_bloku,bajt
G adres
HB adr_pocz,bajt[,N]
HW adr_pocz,słowo[,N]
L adr_pocz[,N]
M adr_źródł,adr_docel,dł_bloku
O#spec_pliku
P adres,bajt
R num_sekt,adr_bufora
T adr_pocz,„tekst"
U adr_pocz[,N]
V adres,słowo
W num_sekt, adr_bufora
Z adr_pocz[,N]

KILKA UWAG O STOSOWANIU SYSTEMU KYAN PASCAL NA ATARI

W uzupełnieniu kilku artykułów, które ukazały się ostatnio w „Bajtku" oraz w „Komputerze" na temat Kyan Pascala pozwolę podzielić się kilkoma dodatkowymi uwagami, jakie nasunęły mi się w czasie dość długiego stosowania tego języka.

1. Prawdziwą wygodę w posługiwaniu się **Kyan Pascalem** mają posiadacze wersji 130 KB komputera, pozwalającej na założenie ramdysku. Umieszczenie tam edytora D8:ED oraz kompilatora D8:PC pozwala w sposób prawie natychmiastowy przechodzić z edycji do kompilacji podczas uruchamiania programu. Gdy uruchamiamy program zawiesi się, naciśnięcie klawiszy RESET i OPTION przenosi system do DUP.SYS i w zdecydowanej większości przypadków ramdysk pozostaje nienaruszony pozwalając na kontynuowanie edycji.

2. Prędkość kompilacji jest porównywalna z systemem **Action!** przy czym **Kyan Pascal** w odróżnieniu od tego ostatniego kompiluje program źródłowy złożony na dysku w program wynikowy składany na dysku. Użycie dyrektywy kompilatora „0" pozwala by program wynikowy i źródłowy znalazły się w różnych stacjach dysków. Kompilator sam określa nazwę programu wynikowego dodając do nazwy programu źródłowego kropkę oraz literę „0". W związku z tym próba skompilowania programu źródłowego o nazwie z przyrostkiem po kropce np. PRINT.I kończy się wymazaniem treści programu źródłowego oraz wystąpieniem błędu wyjścia/wejścia.

3. Prędkość działania programu wynikowego jest porównywalna z **Turbo Basicem XL** a więc dwukrotnie mniejsza niż w **Action!**. Autor w oparciu o [1] wykonał tekst prędkości opisany w [2] i otrzymał wynik równy 8567 taktów, co jest bardzo zbieżne z szacunkami Wojciecha Zientary.

4. **Kyan Pascal** zdaniem autora góruje nad **Action!** czytelnością programu źródłowego, możliwością zagnieżdżania typów oraz łatwiejszym przekazywaniem parametrów do procedur i funkcji. Poza tym program wynikowy po połączeniu ze zbiorem LIB może być uruchamiany bezpośrednio z poziomu DUP.SYS.

5. Obsługa błędów jest wystarczająco bogata, ale po wniesieniu poprawek opisanych w [3].

Poniższe uwagi związane są z różnicami omawianej wersji z jego standardem [4] bądź wersją TURBO [6].

1. Typy okrojone i wyliczeniowe, nawet gdy ich moc jest mniejsza od 256, zajmują dwa bajty a nie jeden.

2. Brak instrukcji CLOSE. Plik zamyka koniec programu lub klawisz BREAK. W związku z tym program może obsługiwać tylko pięć plików.

3. W nagłówku programu nie jest wymagana lista plików zewnętrznych. Program komunikuje się z nimi poprzez wykonanie procedury RESET lub REWRITE.

Na przykład wykonanie procedury:

```
reset (g,'D1:PLIK');
```

daje taki sam efekt jak wykonanie ciągu instrukcji w **Turbo Pascalu:**

```
assign (g,'D1:PLIK');
reset (g);
```

4. Wykonanie ciągu instrukcji:

```
reset (g,'D1:PLIK');
seek (g,n);
```

otwiera zbiór o nazwie D1:PLIK zarówno do odczytu jak i zapisu. Zbiór ten nie może być zabezpieczony.

5. Po słowie kluczowym WITH wymagana jest nazwa zmiennej rekordowej a nie zmienna rekordowa. Nie jest ograniczony typ zmiennej rekordowej.

6. Przed użyciem instrukcji GRAPHICS(8) należy RAMTOP ustawić na wartość 96 lub nawet 128 stosując procedurę POKE (106.0,128). W tym przypadku pamięć ekranu zajmuje komórki od 24576 do 32768. Poniżej aż do komórki 8192 może zawierać się program.

7. Każdy program powinien kończyć się przejściem do trybu tekstowego poprzez wykonanie ciągu instrukcji:

poke (106.0,192);
Graphics (0);

W przeciwnym wypadku mogą wystąpić kłopoty na przykład z wczytaniem edytora bądź kompilatora.

W uzupełnieniu podaję kilka procedur przeznaczonych do włączania dyrektywą #I:

1. **STRINT** — zamiana zmiennej typu integer w zmienną tekstową. Rekurencyjna wersja wewnętrznej procedury WSTAW powstała na podstawie przykładu w [6]. Użycie procedury STRINT wymaga wcześniejszej definicji:

TYPE tekst = ARRAY [1..15] OF char

2. **TEXT8GR** — umieszczenie na ekranie w trybie 8 pierwszych LEN znaków tablicy tekstowej o nazwie W począwszy od miejsca o współrzędnych X oraz Y. Współrzędne X oraz Y jak dla trybu tekstowego 0. Wymaga wcześniejszej definicji typu TEKST jak w punkcie 1 oraz określenia funkcji i procedury POKE oraz PEEK. Użycie instrukcji IF koryguje kolejność znaków w zestawie ROM z kolejnością w kodzie ATASCI.

3. **PUTEKRAN** — pozwala zrzucić pamięć ekranu w trybie 8 na dysk. Procedura ta ingeruje w 1 kanał IOCB, zachowuje format zapisu stosowany w programach firmy XLEnt,

4. **COPY** — pozwala utworzyć kopię zbioru. Powinna zainteresować przede wszystkim posiadaczy RAMDISK. Ponieważ kopiowane pliki są typu char konieczne było użycie instrukcji IF, aby program nie tylko rozpoznawał ale również przepisywał znak końca wiersza: ord()=155

Wymienione procedury są z powodzeniem stosowane przez autora, czego życzy on wszystkim użytkownikom **Kyan Pascal** na małe Atari.

Ryszard Wiech

Cytowana literatura

1. Wiech R: „Peek i „Peek i poke w KYAN PASCALU", BAJTEK, 1988, nr 12
2. Zientara W: „ACTION! znaczy szybkość", BAJTEK, 1988, nr 4
3. Zabołotny W: „Korekta KYAN PASCALA", BAJTEK, 1988, nr 4
4. Iglewski M......: „PASCAL język wzorcowy. Pascal 360", WNT, 1984
5. Bielecki J: „TURBO PASCAL. Wersja 3.0", WNT, 1988
6. Kernighan B. Ritichie D: „Język C", WNT, str. 102.

Od redakcji: Action! także posiada możliwość kompilacji z dyskietki, ale jedynie do pamięci (zob. pkt 2). Istnieje plik ACTION. LIB, który umożliwia działanie programu bez obecności kompilatora (zob. pkt 4).

LISTING 1

```
PROCEDURE strint(wartosc:integer;VAR postac:tekst);

 VAR
  szerokosc,l:integer;

PROCEDURE wstaw(miejsce,liczba:integer);
 VAR
  i:integer;
 BEGIN
  i:=trunc(liczba/10);
  IF i<>0
   THEN
    wstaw(miejsce-1,i);
  postac[miejsce]:=chr((liczba MOD 10)+48)
 END;

BEGIN
 IF wartosc=0
  THEN
   BEGIN
```

```
    postac[1]:='0';
    szerokosc:=1;
   END
  ELSE
   BEGIN
    szerokosc:=trunc(ln(obs(wartosc))/ln(10)+1);
    IF wartosc<0
     THEN
      BEGIN
       postac[1]:='-';
       wartosc:=-wartosc;
       szerokosc:=szerokosc+1;
      END;
    wstaw(szerokosc,wartosc)
   END;
 FOR I:=szerokosc+1 TO 15 DO
  postac[I]:=' '
END;
```

LISTING 2

```
PROCEDURE copy(n1,n2:tekst);
 VAR
  f,g:FILE OF char;
 BEGIN
  rewrite(f,n2);
  reset(g,n1);
  WHILE not eof(g) DO
   BEGIN
    IF not eoln(g)
     THEN
      f^:=g^
     ELSE
      f^:=chr(155);
    put(f);
    get(g)
   END
 END;
```

LISTING 3

```
PROCEDURE putEkran(nazwa:tekst)

 VAR
  p:FILE OF char;

 PROCEDURE close;
  BEGIN
   BEGIN
#a
   TXA
   PHA
   LDX #16
   LDA #12
   STA 850
   JSR 58454
   PLA
   TAX
#
   END
 END;

 BEGIN
  close;
  rewrite(p,nazwa);
  BEGIN
#a
  TXA
  PHA
  LDX #16
  LDA #11
  STA 850
  LDA 88
  STA 852
  LDA 89
  STA 853
  LDA #0
  STA 856
  LDA #30
  STA 857
  JSR 58454
  PLA
  TAX
#
  END;
  close
 END;
```

LISTING 4

```
PROCEDURE text8Gr(w:tekst;len,X,Y:integer);

 VAR
  a,b:integer;
  d1,d2,d3,d4:real;

 BEGIN
  d1:=peek(560,0)+256*peek(561,0);
  d2:=peek(d1+4)+256*peek(d1+5);
  FOR a:=1 TO len DO
   BEGIN
    IF ((ord(w[a])<=96 AND (ord(w[a])>=32))
     THEN
      d3:=57344+8*(ord(w[a])-32)
     ELSE
      d3:=57344+8*ord(w[a]);
    d4:=d2+320*Y+X+a-1;
    FOR b:=0 TO 7 DO
     poke(d4+40*b,peek(d3+b))
   END
 END
```

POWIERZCHNIA OBSZARU NIEREGULARNEGO

Konieczność obliczenia powierzchni nieregularnej figury występuje często w praktycznych zastosowaniach. Pomocne może się tu okazać zastosowanie do obliczeń komputera.

Pierwszym krokiem jest przedstawienie mierzonego obszaru w postaci zrozumiałej dla maszyny. Będą to współrzędne XY kolejnych punktów ograniczających mierzony obszar. Dane te muszą być podawane z zachowaniem kolejności punktów, natomiast kierunek może być dowolny: zgodnie lub przeciwnie do ruchu wskazówek zegara.

Dokładność uzyskanego wyniku zależy od liczby punktów użytych w obliczeniach. Dla obszarów ograniczonych linią złożoną z prostych odcinków należy podać współrzędne wszystkich wierzchołków. Nieregularna linia ograniczająca obszar musi być najpierw przedstawiona w postaci krótkich odcinków. Zagęszczenie tych odcinków powinno być dobrane w zależności od krzywizny linii oraz wymaganej dokładności wyniku.

ALGORYTM

Popatrzmy na rysunek 1. Pokazany jest tam trójkąt OAB. Jego powierzchnia jest równa sumie powierzchni trójkąta OCB i czworokąta ABCD zmniejszonej o powierzchnię trójkąta ODA:

pole (OAB) = pole(OCB) + pole(ABCD) − pole(ODA)

Zapiszmy to równanie matematycznie oznaczając poszukiwaną powierzchnię trójkąta OAB jako S:

$$S = X2*Y2/2 + (X1-X2)*(Y1+Y2)/2 - X1*Y1/2$$

Teraz wykonamy kolejne przekształcenia w celu uproszczenia otrzymanego wzoru:

$$S = (X2*Y2 + X1*Y1 + X1*Y2 - X2*Y1 - X2*Y2 - X1*Y1)/2$$

$$S = (X1*Y2 - X2*Y1)/2$$

Ponieważ znak tak osiągniętego wyniku zależy od kolejności numerowania punktów, to ostatecznym rezultatem będzie wartość bezwzględna S.

Teraz popatrzmy na czworokąt ABCD pokazany na rysunku 2. Jego powierzchnia jest równa sumie powierzchni trójkątów OAB, OBC i OCD minus powierzchnia ODA. Wykorzystując uzyskany wcześniej wzór na obliczenie powierzchni trójkąta otrzymamy:

$$S = (X0*Y1 - X1*X0 + X1*Y2 - X2*Y1 + X2*Y3 - X3*Y2 + X3*Y0 - X0*Y3)/2$$

a po podstawieniu wartości liczbowych:

$$S = (10*40 - 30*40 + 30*30 - 40*40 + 40*10 - 40*30 + 40*40 - 10*10)/2$$

$$S = (400 - 1200 + 900 - 1600 + 400 - 1200 + 1600 - 100)/2$$

$$S = -800/2 = -400$$

$$S = ABS(S) = 400$$

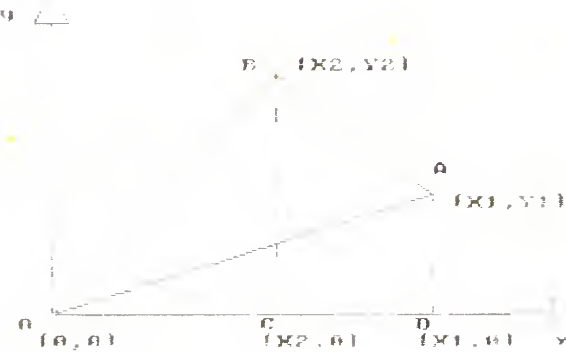
Obliczenie powierzchni dowolnego obszaru ograniczonego zamkniętą linią będzie przebiegało identycznie, lecz dla innej liczby punktów. Ogólnie dla n punktów ponumerowanych od 0 do n-1 otrzymamy wzór:

$$S = \frac{1}{2}\left|\sum_{k=0}^{n-1}(x_k y_{k+1} - x_{k+1} y_k)\right|$$

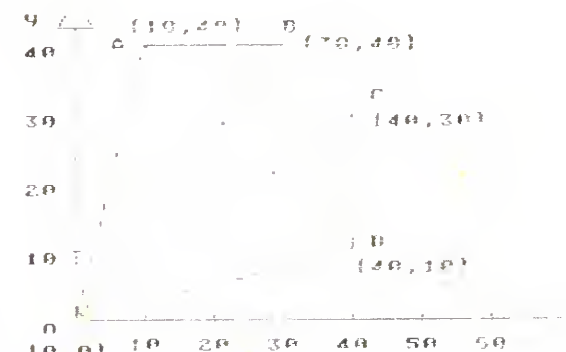
PROGRAM

Napiszmy teraz program, który będzie realizował obliczenia według uzyskanego algorytmu. Aby program ten był uniwersalny, współrzędne wszystkich punktów zostały zapisane w instrukcji DATA (wiersz 260). Trzeba przy tym pamiętać, aby punkt 0 został zapisany jako ostatni. Ponadto dodatkowa instrukcja DATA w wierszu 250 zawiera liczbę punktów użytych do obliczeń. Program umożliwia wykorzystanie do 50 punktów, lecz po zmianie wymiarów w deklaracjach tablic liczbowych (wiersz 100) mozna liczbę punktów zwiększyć. Obliczona powierzchnia jest przypisywana zmiennej S i po rozbudowaniu programu może być użyta do dalszych obliczeń.

Wojciech Zientara



Rys. 1



Rys. 2

```
YS  10 REM POWIERZCHNIA OBSZARU
FN  20 REM Wojciech Zientara
DA  30 REM Copyright (C) Bajtek
BB  40 REM
UZ  100 DIM X(50),Y(50)
KV  110 RESTORE 250:READ N
DT  120 FOR K=1 TO N
PO  130 READ A,B
NU  140 X(K)=A:Y(K)=B
GV  150 NEXT K
NE  160 X(0)=X(N):Y(0)=Y(N)
HO  170 S=0
PD  180 FOR K=0 TO N-1
WF  190 S=S+X(K)*Y(K+1)-X(K+1)*Y(K)
GM  200 NEXT K
YP  210 S=ABS(S)/2
NY  220 ? "Powierzchnia wynosi ";S
NX  230 END
UO  240 REM liczba i wspolrzedne punktow
OG  250 DATA 4
YI  260 DATA 40,30,40,10,10,40,30,40
```

NASZ PRYWATNY KONKURS

Przygotowanie listingu programu do druku w „Bajtku" polega na sprawdzeniu poprawności jego działania i obliczeniu kodów kontrolnych dla „Edytora Basica". Druga z tych czynności jest wykonywana przez specjalny program. Niestety dopisanie kodów do programu odbywa się „ręcznie" i jest wyjątkowo nieciekawym zajęciem.

Robiąc to kolejny raz wpadliśmy na pomysł ogłoszenia konkursu. Będzie to nasz prywatny konkurs dla wszystkich, którzy próbują pisać programy w Basicu. Zadanie jest proste: trzeba napisać krótki program w Basicu, musi on jednak spełniać trzy wymienione niżej warunki:

1. Program nie może być krótszy niż 10 wierszy. Liczba instrukcji w każdym wierszu jest dowolna, lecz zabronione jest używanie komentarzy (REM).
2. Program musi mieć sens. Jest to pojęcie względne i mogą być kłopoty z jego interpretacją. W każdym przypadku decyzja będzie podjęta przez jury.
3. Kody „Edytora Basica" służące do kontroli poprawnego przepisania programu muszą:
   a) być jednakowe lub
   b) tworzyć jakiś napis

Oczywiście żaden program nie może spełniać obu tych warunków jednocześnie, więc konkurs będzie rozgrywany w dwóch konkurencjach.

Odpowiedzi prosimy nadsyłać do redakcji „Bajtka" (ul. Wspólna 61, 00–687 Warszawa) koniecznie z dopiskiem na kopercie „PRYWATNY KONKURS", do dnia 30 września 1989.

Nagrodą dla twórców odpowiedzi będzie opublikowanie ich programów w „Bajtku", a autorzy otrzymają wynagrodzenie zgodnie z obowiązującymi stawkami.

Na Wasze listy oczekuje jury w składzie:

Marek Zachar
Wojciech Zientara

JĘZYK MASZYNOWY

cz. IV

Zanim rozwiążemy zadanie z poprzedniego wykładu, chciałbym przedstawić sposób na zapamiętanie naszych coraz dłuższych programów na taśmie lub dysku. Wcześniej czy później stanie się to koniecznością, gdyż szkoda nam będzie cennego czasu na kilkukrotne wstukiwanie tego samego programu, gdy z jakichś względów zechcemy do niego powrócić.

Metoda jest prosta i najlepiej nadaje się do programów o objętości do kilkudziesięciu bajtów, czyli programów, jakie my piszemy. Oczywiscie przedstawioną metodę stosować można do większych programów, ale w tym wypadku łatwiej będzie chyba skorzystać z gotowego programu np. DATAMAKER zamieszczonego w numerze specjalnym Bajtek Tylko o Commodore. Załóżmy, że nasz program zawiera się między adresami $2710 i $2729 (dziesiętnie 10000 i 10025). Wprowadzamy w BASIC-u, w trybie bezpośrednim następującą linię: FOR I=10000 TO 10025.PRINT PEEK (I);:NEXT I. Przedstawiona pętla spowoduje wyświetlenie na ekranie w notacji dziesiętnej kolejnych bajtów naszego programu. Program wygląda teraz inaczej niż gdybyśmy przeglądali ten sam obszar pamięci za pomocą monitora języka maszynowego i instrukcji M 2710 2729, ale nadal jest to ten sam program. Teraz otrzymane liczby przekształcamy posługując się edytorem ekranowym w linie BASIC-owe zaczynające się od numeru linii i instrukcji DATA. W dalszej części dzisiejszego wykładu zamienimy program z naszej pracy domowej na program napisany w BASIC-u nadający się do zapamiętania na dysku lub taśmie. Mając już tak przygotowane linie wystarczy dopisać kolejne linie w języku BASIC: nr linii FOR I=10000 TO 10025:READ X:POKE I, X:NEXT I. Działania takiego programu chyba tłumaczyć nie muszę gdyż większość programów zamieszczanych w Bajtku jest właśnie takiej postaci. Oczywiście należy w trakcie dokonywanych operacji zadbać o to by program był lokowany w tym samym obszarze pamięci, z którego został pobrany. W tym celu adresy (u nas 10000 i 10025) w programie, jak i w linii programu wykonywanej w trybie bezpośrednim muszą być identyczne. Do programu można dodać kolejną linię z instrukcją SYS 10000, która spowoduje uruchomienie naszego programu maszynowego. Tak przygotowany program w języku BASIC możemy w klasyczny sposób zgrać na taśmę instrukcją SAVE„nazwa" lub na dysk instrukcją SAVE„nazwa",8 lub DSAVE„nazwa" Wkrótce poznamy sposób na zgrywanie naszych programów z pominięciem przechodzenia etapu języka BASIC. Nastąpi to jednak po wprowadzeniu takich pojęć jak początek obszaru zmiennych, początek i koniec programu itp.

Przejdźmy teraz do rozwiązania naszej pracy domowej LISTING 1. Podczas ostatniego wykładu zapoznaliśmy się z pętlami, które w tym programie zostały intensywnie wykorzystane co spowodowało, że program jest w miarę krótki i przejrzysty. Pierwsza część programu powoduje wyczyszczenie ekranu (odpowiednik PRINT CHR$(147)) i wyświetlenie komunikatu „PODAJ CYFRE (1–9)" Jak pamiętamy w celu wyświetlenia informacji na ekranie posługujemy się procedurą CHROUT zaczynająca się od adresu $FFD2. W celu wyświetlenia napisu posługujemy się adresowaniem ideksowanym. Rejestrem indeksującym jest rejestr .X, który zmienia wartości od #$0 do #$13. Pisząc ten program umieściłem wszystkie napisy razem, jako dane, tuż za ostatnią instrukcją programu. Pisząc program musiałem jedynie pamiętać od którego adresu zaczynają się kolejne napisy i ile zawierają znaków. Pierwszy napis zaczyna się od adresu $2741 i ma 19 bajtów ($13 szesnastkowo). Spytacie się zapewne dlaczego więc rejestr X zmienia swoją zawartość od $0 do $13 (czternaście wartości). Otóż wewnątrz pętli (kończącej się lub jak kto woli zaczynającej się instrukcji BNE $2712) jest instrukcja INX powodująca powiększanie zawartości rejestru .X o jeden. Kolejna instrukcja to instrukcja porównania zawartości .X z podaną wartością tu #$13 (CPX #$13). Gdybym w tym miejscu podał #$12 to kolejna instrukcja BNE $2712 spowodowałaby przejście do kolejnej linii programu i nie zostałby wyświetlony ostatni znak napisu.

Oprócz procedury CHROUT posłużyłem się w programie nową procedurą o nazwie GETIN wywoływanej instrukcją JSR $FFE4. Procedura ta powoduje przesłanie znaku z kanału Wejścia do rejestru .A. Kanałem wejścia jest w naszym przypadku bufor klawiatury. Znak przedstawiony jest za pomocą kodu ASCII. Działanie tej procedury jest bardzo podobne do działania instrukcji GET w języku BASIC z tą różnicą, że wprowadzany znak nie jest wyswietlany (chyba, że od razu po wprowadzeniu znaku skorzystamy z procedury CHROUT) Ponadto procedura nie czeka na wciśnięcie klawisza lecz powraca od razu do programu wywołującego. Jeżeli bufor klawiatury był pusty to zawartość .A wyniesie 0. W opisie procedury czytamy, że oprócz zmian zawartości .A, mogą zaistnieć przypadki, w których zmianie ulegają zawartości rejestrow .X i Y. Należy zatem zadbać by przed wywołaniem procedury GETIN wszelkie ważne dla działania programu informacje zostały zapamiętane w jakiś dwóch komórkach pamięci.

Druga część programu zaczyna się właśnie od skoku do procedury GETIN. Wczytana zostaje tu (ze sprawdzeniem poprawnosci) wartość numeryczna z przedziału 1–9. Wprowadzona cyfra wskazuje ile razy chcemy by pojawił się nasz napis Ponieważ dla cyfr 1–9 kody ASCII zawierają się w przedziale $31–$39 należy sprawdzić czy zawartość A (wczytana cyfra) znajduje się w tym przedziale. Poprzednio w tym cyklu wykładów jak i w cyklu poświęconym monitorom mowa była o instrukcjach z serii porównań i rozgałęzień (instrukcji decyzyjnych). W tym miejscu chciałbym przedstawić pewne rozszerzenie wcześniej zawartego opisu Chodzi o to jak sprawdzić warunki „mniejszy" lub „większy równy" Nie ma problemu jeśli chodzi o sprawdzenie równości (znamy juz instrukcje BNE i BEQ). Natomiast jeśli chodzi o znacznik C to do tej pory mówiliśmy o nim wyłącznie jako o znaczniku przeniesienia nie wnikając dogłębnie w sposób jego działania. W odróżnieniu od znacznika np. Z,, na który wpływ ma większość instrukcji, na znacznik C wpływ mają wyłącznie instrukcje porównań (CPX, CPY, CMP) oraz instrukcje arytmetyczne. Porównanie polega na odjęciu porównywanych wartości i sprawdzeniu czy wynik jest dodatni czy ujemny. Jak z tego wynika znacznik C odpowiada nam na pytanie czy porównywane liczby spełniają warunek „mniejszy" czy „większy równy" W działaniach arytmetycznych wskazania znacznika C są takie same jak omówione do tej pory we wcześniejszych wykładach. A więc instrukcja BCC sprawdza warunek „mniejszy" (znacznik C ustawiony na 0) a instrukcja BCS sprawdza warunek „większy równy" (znacznik C ustawiony na 1). Znamy już sposób na sprawdzanie poprawności wprowadzanych argumentów. Dzięki tym nowym warunkom sprawdzimy czy cyfra należy do określonego przedziału bez konieczności porównywania wprowadzanych wartości z każdą z liczb z zakresu $31–$39. Pierwsza instrukcja sprawdza czy wprowadzona cyfra nie jest mniejsza od $31 (odpowiednik 1). Jeżeli tak jest to następuje powtórny skok (BCC$271D) do procedury GETIN. Zauważmy mimo tego, że procedura nie czeka na wciśnięcie klawisza to jeżeli .A zawiera wartość 0 to powyzszy warunek również nie będzie spełniony i program tak długo będzie powracać do procedury GETIN aż zostanie wprowadzona pewna wartość z klawiatury. Jeżeli już się okaże, że wprowadzona wartość jest większa od $31 to następnie należy sprawdzić czy jest ona mniejsza/równa od $39 lub inaczej czy nie jest większa od $39 Ponieważ instrukcja BCS sprawdza warunek „większy równy" dlatego też zawartość. A porównamy z kodem $3A a nie $39. Jeżeli się już okaże, że wprowadziliśmy do komputera cyfrę to zawartość .A zostanie przesłana do .Y (dla wprawy w posługiwaniu się instrukcjami transferu) i nastąpi wyczyszczenie ekranu Teraz pamiętając już od jakiego adresu zaczyna się drugi napis ($2754) i ile ma bajtów (39), i ile razy ma być wyświetlony pełny napis (zawartość .Y) przystępuję do jego wyświetlania. W ostatniej części programu zagnieżdżone są wewnątrz siebie dwie pętle. Wewnętrzna tak jak w pierwszej części powoduje w oparciu o adresowanie indeksowane wyświetlenie napisu o określonej długości, zewnętrzna kontroluje zaś liczbę wyświetlonych napisów. Koniec programu oznaczamy jak zwykle instrukcją BRK lub RTS jeżeli

program ma być wywoływany jako podprogram przez inny program np. przez BASIC.

LISTING 2 przedstawia komplet danych do naszego programu (zakodowane w ASCII nasze napisy i komunikaty). Wartości te należy wprowadzić do kolejnych komórek pamięci bezpośrednio po ostatniej instrukcji programu. Jeżeli Wam się to nie uda to wpiszcie program z LISTING-u 3, uruchomcie go i zobaczcie korzystając z monitora jak należy to zrobić. LISTING 3 jednocześnie pokazuje jak wygląda nasz program w postaci przygotowanej do zgrania na nośnik pamięci. Sprawdzcie, czy potraficie na podstawie LISTING-u 1 stworzyć taki program.

Instrukcjami porównań, odgałęzień oraz znacznikami będziemy się często zajmować w przyszłości dlatego tyle miejsca razem z Klaudiuszem poświęcamy temu tematowi. Na koniec omawiania tego bloku tematycznego odpowiem na pytanie jak sobie poradzić z liczbami ujemnymi skoro niby mamy do dyspozycji znacznik ujemności, ale jedynymi wartościami jakie możemy przechowywać w komórkach to wartości dodatnie z przedziału 0–255. Jak się okazuje w tym samym bajcie można przechować liczby z zakresu −128–+127 (również 256 kombinacji). Bajty fizycznie nadal będą zawierać tę samą kombinację bitów lecz ich interpretacja pozostawiona zostaje programiście. Matematycznie można to wyjaśnić następująco: liczby ze znakiem przechowywane są w formie uzupełnienia do dwóch. Przechowujemy zatem −1 jako $FF, −2 jako $FE itd. aż do −128 jako $80. Jak pewnie zauważyliście liczby heksadecymalne z zakresu $80–$FF mają najstarszy bit ustawiony na 1, a w związku z tym znacznik ujemności wskazuje, że bit ten jest ustawiony a liczba może być traktowana przez programistę jako ujemna (może lecz nie musi). Można rozważyć dziesiętną analogię np. niech licznik naszego magnetofonu wskazuje 0025. Postanawiamy cofnąć taśmę o 30 jednostek. Bylibyśmy zdziwieni gdybyśmy w wyniku tego odejmowania otrzymali −5 natomiast liczba 9995 wydaje się być w tym miejscu czymś absolutnie naturalnym.

Kolejnym zagadnieniem, które dziś zaczniemy omawiać a podczas kolejnych wykładów pogłębiać są operacje logiczne AND, ORA i EOR. Tych trzech instrukcji używamy do kontroli bitów w bajcie. Niezwykła rola tych instrukcji polega na tym, że wpływ możemy mieć na każdy z bitów niezależnie od konfiguracji pozostałych bitów. Do czego może być to przydatne? Okazuje się, że są takie operacje w których komputer nie traktuje bajtu jako liczby lecz właśnie jako zbiór ośmiu bitów, z których każdy informuje go o czymś innym np. w portach wejścia/ /wyjścia każdy bit może kontrolować inny sygnał. Możemy więc żądać włączenia jednego z sygnałów bez zmiany pozostałych. Ponadto wiele efektów audiowizualnych może być osiągnięte tą drogą a więc chyba warto poznać reguły rządzące operacjami logicznymi.

Wszystkie instrukcje logiczne działają wyłącznie w oparciu o rejestr .A (akumulator). Coraz częściej okazywać się będzie, że .A jest rejestrem uprzywilejowanym, w którym to tylko i wyłącznie dokonuje się operacji logicznych, arytmetycznych, do którego lub z którego przesyłane są informacje.

W skrócie AND to koniunkcja lub iloczyn logiczny i służy do wyłączania bitów (ustawiania ich na 0). ORA to alternatywa lub suma logiczna i służy do włączania bitów (ustawiania ich na 1). EDR to LUB wykluczające służące do przełączania bitów-z 1 na 0 i odwrotnie.

AND — dla każdego bitu w .A instrukcja działa następująco:

bit	maska	wynik
0	0	0
1	0	0
0	1	0
1	1	1

Zauważmy, że gdy maska jest 0 to wynik jest również 0. Gdy maska wynosi 1 to oryginalny bit nie ulega zmianie. Zatem jak widać AND służy wyłączaniu bitów np. aby wyłączyć bity 4,5,6 w $C7 wykonujemy $C7 AND $8F

```
wartość $C7 = 11000111
maska   $C7 = 10001111
wynik         10000111
               xxx
```

ORA — dla każdego bitu w .A instrukcja działa następująco:

bit	maska	wynik
0	0	0
1	0	1
0	1	1
1	1	1

Jeżeli maska wynosi 0 to oryginalny bit .A nie ulega zmianie, gdy maska wynosi 1 to oryginalny bit jest włączany np. włącz bity 4,5,6 w $C7. Robimy to następująco $C7 ORA $70

```
wartość $C7 = 11000111
maska   $70 = 01110000
wynik         11110111
               xxx
```

EOR — dla każdego bitu w .A instrukcja działa następująco:

bit	maska	wynik
0	0	0
1	0	1
0	1	1
1	1	0

Jeżeli maska wynosi 0 to nic nie ulega zmianie natomiast przy masce 1 0 zmienia się na 1 a 1 na 0 np. przełącz bity 4,5,6 w $C7. $C7 EOR $70

```
wartość $C7 = 11000111
maska   $70 = 01110000
wynik         10110111
               xxx
```

Za miesiąc ciąg dalszy

Dominik Falkowski

LISTING 1

```
READY.

MONITOR
    PC  SR AC XR YR SP
;
B000 00 00 00 00 F8

 02710  A2 00     LDX #$00
 02712  BD 41 27  LDA $2741,X
 02715  20 D2 FF  JSR $FFD2
 02718  E8        INX
 02719  E0 13     CPX #$13
 0271B  D0 F5     BNE $2712
 0271D  20 E4 FF  JSR $FFE4
 02720  C9 31     CMP #$31
 02722  90 F9     BCC $271D
 02724  C9 3A     CMP #$3A
 02726  B0 F5     BCS $271D
 02728  A8        TAY
 02729  A9 93     LDA #$93
 0272B  20 D2 FF  JSR $FFD2
 0272E  A2 00     LDX #$00
 02730  BD 54 27  LDA $2754,X
 02733  20 D2 FF  JSR $FFD2
 02736  E8        INX
 02737  E0 27     CPX #$27
 02739  D0 F5     BNE $2730
 0273B  88        DEY
 0273C  C0 30     CPY #$30
 0273E  D0 EE     BNE $272E
 02740  00        BRK
```

LISTING 2

```
>0274  93 [illegible]
>02749 [illegible]
>02751 [illegible]
>02759 50 [illegible]
>02761 [illegible] 4A [illegible]
>02769 [illegible] 20 [illegible]
>02771 [illegible]
>02779 45 8D [illegible]
```

LISTING 3

```
10 DATA A2,00,BD,41,27,20,D2,FF,E8,E0,13,D0,F5,20,E4,FF
20 DATA C9,31,90,F9,C9,3A,B0,F5,A8,A9,93,20,D2,FF,A2,00
30 DATA BD,54,27,20,D2,FF,E8,E0,27,D0,F5,88,C0,30,D0,EE
40 DATA 00,53,50,4F,44,41,44,20,43,59,46,52,45,20,28,31
50 DATA 2D,39,29,3D,4E,41,4A,4C,45,50,53,5A,59,20,57,20
60 DATA 42,41,4A,54,4B,55,20,44,45,53,54,20,4B,4C,41,4E
70 DATA 20,43,4F,4D,4D,4F,44,4F,52,45,3D,00,00
80 FOR J= 10000 TO 10103:READ D$:POKE J,DEC(D$):NEXT
```

LISTA PROGRAMÓW UŻYTKOWYCH

Nie wszystkie programy, na które głosują Czytelnicy, są programami najnowszymi, nie wszystkie z nich są aktualnie powszechnie dostępne (np. Timoworks DTP dla C-128).

Po otrzymaniu pierwszych propozycji słownych i listownych zdecydowaliśmy się na wprowadzenie nowej rubryki przeznaczonej dla programów z serii DeskTop Publishing, co w wypadku Commodore oznacza głównie programy umożliwiające łączenie grafiki i tekstu w dowolnym układzie.

Z prawdziwych nowości warto wspomnieć o programie AMICAPAINT na który głosowało sporo Czytelników. Z ciekawostek warto wspomnieć o module WARP SPEED pracującym w obu trybach pracy — C-128 i C-64, co niewątpliwie jest wielkim ułatwieniem dla użytkownika. Dużo głosów było również za modułem FINAL CARTRIGE III; zapewne ucieszę Czytelników informacją, że niebawem BAJTEK przedstawi porównanie najpopularniejszych w kraju kart tego typu.

(kd+df)

TYP PROGRAMU	COMMODORE 128	OSPRZET	COMMODORE 64	OSPRZET
Kalkulacyjny	SWIFTCALC 128	D	=> CALC RESULT	D
Baza danych	=> DATA MANAGER 128	D	MANAGER 64	D
Edytor tekstu	=> STAR TEXTER 128	D	=> GEOWRITE V2.0	DJ(A)
Grafika	=> ULTRAHIRES	D	=> AMICAPAINT	DJ
Grafika 3D	CAD 3D	DJ	GIGA CAD PLUS	DJ
Desktop Publ.	=> TIMEWORKS DTP	D	=> PRINTFOX	DJ
Muzyka	MUSIC MAKER 128	D	ROCK MONITOR V5.0	D
Asembler	LADS	D	CBM MACROASSEMBLER	D
System	GEOS 128	DJ(A)	GEOS V1.3	DJ(A)
Rozszerz. BASIC	METABASIC 128	D	SIMON'S BASIC	
Kompilator	BASIC 128	D	BASIC 64	D
Narz. dyskowy	DOS SHELL	D	=> DISK TOOL	D
Kopiujacy	FAST HACK'EM V4.1B	D	FAST HACK'EM V4.1B	D
Matematyka	-		MATHEMAT	D
TURBO (tasma)	-		TURBO ROM	T
TURBO (dysk)	=> WARP SPEED (modul)		SUPERDOS V2.0	D
Modul	-		=> FINAL CARTRIDGE 3	
Pakiet	GEOS	DJ(A)	GEOS	DJ(A)

MONITORY ML cz. 6

Jeżeli zbadałeś wynik, jaki otrzymaliśmy w komórce 2720 po wykonaniu programu podanego w uprzedniej części cyklu, to prawdopodobnie byłeś lekko zdziwiony — od kiedy 128+128 jest równe 0?

W normalnej arytmetyce „papierkowej" nie używamy pojęcia bitów czy bajtów — ograniczenie to jednak ma komputer. Powiedzieliśmy już wcześniej, że maksymalna liczba jaką można zapisać w pojedyńczej komórce pamięci wynosi 255 — tyle mieści się w 8 bitach. W momencie zapisywania tej wartości do komórki pamięci mikroprocesor wie już, że liczba ta jest zbyt wielka; dlatego też wpisuje tam liczbę 0, a informację „liczba zbyt duża" przekazuje użytkownikowi poprzez zmianę stanu logicznego znacznika przeniesienia w rejestrze słowa stanu. Spróbujmy teraz zabawić się w małą zgaduj-zgadulę za pomocą następującego programu:

```
CLC
LDA    ....
ADC    ....
BRK
```

W miejsce kropek wstaw dowolne adresy z zakresu $0000 do $FFFF. Następnie spróbuj określić na podstawie rejestru słowa stanu czy suma liczb zawartych w tych komórkach była większa od 255 czy nie. Jeżeli znacznik przeniesienia (C) przyjmie stan logiczny 1 to na pewno suma tych liczb jest większa od 255. Sprawdź to empirycznie np. za pomocą polecenia M lub funkcji PEEK w BASIC.

Być może zastanawiasz się dlaczego z naszego programu zniknęła instrukcja STA $2720. To proste. Wynik dodawania jest zapisywany bezpośrednio w akumulatorze; ponieważ po wykonaniu polecenia BRK na ekranie pojawia się nagłówek monitora bardzo łatwo możesz sprawdzić liczbę w nim zawartą (pod skrótem AC.).

Zanim przejdziemy do omawiania następnych znaczników posłużymy się pewnym trickiem mającym na celu zerowanie rejestru słowa stanu. Wpisz następujący program:

```
2750  LDA   #$00
2752  PHP
2753  PHA
2754  PLP
2755  PLA
2756  BRK
```

Programik działa następująco: po wpisaniu do akumulatora wartości 0 (wszystkie bity wyzerowane) zapisujemy akumulator na stosie zawartość rejestru słowa stanu (PHP), a następnie zawartość akumulatora (PHA). Teraz wartości te odczytujemy, dzięki czemu ostatnia wpisana liczba (czyli $00) zostanie przeniesiona do rejestru słowa stanu, a jego poprzednia zawartość — do akumulatora. Teraz przed każdym podanym niżej programem wykonaj G 2750 lub rozpoczynaj każdy, program od JSR $2750, przy czym w tym ostatnim wypadku należy wpisać zamiast BRK — RTS na końcu procedury.

W naszych rozważaniach powinniśmy uwzględnić fakt, że bit piąty rejestru .P, jest niewykorzystywany, stąd też zmiany jego stanu możemy zignorować. Podczas testowania ustawiony będzie również znacznik B, co jest zupełnie naturalne — na końcu każdego programu wykonywana jest zawsze instrukcja BRK powodująca ustawienie tego bitu. Najbardziej interesujące będą więc dla nas znaczniki N, V, Z i C. Dwa pozostałe (D oraz I) przyjmują zwykle stan logiczny 1 po wykonaniu instrukcji SED i SEI, co powoduje albo włączenie trybu liczenia dziesiętnego (a właściwie dziesiętnego kodowanego binarnie — BCD), albo też wyłączenie przerwań IRO. O znacznikach tych porozmawiamy nieco później.

Znaczniki N i V są przeznaczone do bardziej skomplikowanych zadań związanych z operacjami przeprowadzanymi na liczbach ze znakiem (np. + 17 czy −9). W takich obliczeniach sama liczba zajmuje tylko 7 pierwszych bitów danego bajtu — są to bity o numerach 0—6. Bit ostatni (siódmy) jest wykorzystywany do zapisu znaku. W arytmetyce binarnej z kolei znacznik N przyjmuje stan logiczny 1, gdy wynik operacji jest ujemny, a jest zerowany, gdy wynik danej operacji wynosi zero. Na jego stan mają również wpływ operacje na liczbach i same liczby, w których bit 7 jest ustawiony, co wiąże się właśnie z arytmetyką liczb ze znakiem. Wykonajmy teraz następujący przykład (odejmowanie liczby 9 od 8 i 8—8):

G 2750

Stan logiczny wyników:

```
2710   SEC          C=1
2711   LDA   #$08
2713   SBC   #$09   C=0,  N=1
2715   BRK          B=1
2716   SEC          C=1
2717   LDA   #$08
2719   SBC   #$08   Z=0
2718   BRK          B=1
```

Pierwsza instrukcja BRK spowoduje zatrzymanie programu w środku — na ekranie ukaże się nagłówek monitora. Jeżeli Twój monitor wyświetla tylko SR i liczbę, to zamień ją teraz na liczbę binarną — przekonasz się, że po odejmowaniu znacznik N przyjął stan logiczny 1. Wykonaj teraz:

G 2716

Powtórna analiza liczby zawartej w rejestrze słowa stanu pokaże Ci, że znacznik ten został wyzerowany, natomiast dwa inne — Z i C — są ustawione. Dlaczego tak się dzieje?

Zastanówmy się najpierw, po co ustawiamy znacznik C przed wykonaniem odejmowania. Cel jest taki sam jak w normalnej arytmetyce. Gdy mamy odjąć np. 9 od liczby mniejszej, to do tej liczby dodajemy zwykle 10; następnie odejmujemy 9 w tym wypadku nie od 8 lecz już od 18. Całość określana jest zwykle „odejmowaniem z pożyczeniem".

Dokładnie w ten sam sposób dzieje się w naszym programie. Znacznik C jest ustawiany na wypadek, gdyby odjemnik był większy od odjemnej — „pożyczenie" następuje ze znacznika C, który (gdy takie pożyczenie będzie konieczne) zostanie po nim wyzerowany. Prześledźmy ten fragment programu:

ETAP 1

Po ustawieniu znacznika przeniesienia (C=1) do akumulatora wpisywana jest wartość 8.

ETAP 2

Odejmowanie liczby 9 od 8 powoduje w pierwszej kolejności ustawienie znacznika N (N=1). Mikroprocesor „wie" już, że odjemnik jest większy niż odjemna.

ETAP 3

Następuje „pożyczenie" ze znacznika C, co powoduje wyzerowanie tego ostatniego.

ETAP 4

Operacja odejmowania zostaje poprawiona, a wynik wpisany do akumulatora.

Zanalizujemy teraz drugą część programu od komórki o adresie $2716. Odejmowanie 8—8 nie wymaga „pożyczenia", gdyż wynik będzie równy zero. Dlatego też znacznik zera (Z) przyjmie stan logiczny 1. Ponieważ pierwszą instrukcją programu jest SEC powodujące ustawienie znacznika C, a pożyczka nie nastąpiła, znacznik ten będzie w rezultacie ustawiony nadal.

Znacznik zera (Z) przyjmuje stan logiczny 1 nie tylko wtedy, gdy wynik operacji jest równy zero. Stan ten przyjmuje on również wtedy, gdy do dowolnego rejestru wczytamy liczbę 0.

```
G 2750
2710   LDX   #$00
2718   BRK
```

Warto tu także przypomnieć pewną „anomalię" — znacznik Z jest ustawiany wtedy, gdy wynikiem operacji JEST zero; stan logiczny „0" przyjmuje on w wypadku przeciwnym.

Do omówienia pozostał nam jeszcze znacznik przepełnienia — V. Jest on wykorzystywany głównie w arytmetyce liczb ze znakiem i służby tu do sygnalizowania przepełnienia bitu 6 czyli ostatniego bitu liczby ze znakiem (bit 7 przechowuje wtedy informacje o znaku). Znacznik V pełni tu rolę identyczną jak znacznik C w arytmetyce liczb binarnych — gdy dana liczba nie da się już zapisać w 8 bitach, to do danego rejestru czy komórki pamięci wpisywane jest 0, a sam znacznik C przyjmuje stan logiczny 1. Analogicznie znacznik V przyjmuje stan logiczny 1 w momencie gdy dana liczba (ze znakiem) nie daje się już zapisać w siedmiu bitach. Wykonajmy następujący przykład:

```
G 2750
2710   LDA   #$80
2712   ADC   #$7F
2714   BRK
```

Po zbadaniu zawartości rejestru słowa stanu okaże się jednak, że znacznik V nie zmienił swojego stanu logicznego. Wszystko jest w porządku — do zapisania liczby $7F (127 dziesiętnie) potrzebujemy tylko 7 bitów. Jednakże już liczba 128 ($80) zmieni na pewno stan tego znacznika. Zamień zatem wartość $7F w komórce $2713 na liczbę $80. Po ukazaniu się nagłówka monitora przekonasz się, że znacznik V jest ustawiony. Oprócz tego zmieniły stan logiczny dwa inne znaczniki — Z i C — z przyczyn dokładnie omówionych powyżej.

Być może zastanawiasz się Czytelniku, dlaczego na cały ten rejestr słowa stanu kładę aż tak wielki nacisk. Powód jest prosty — instrukcje warunkowe, takie jak BNE czy BCS, są ściśle uzależnione od aktualnego stanu poszczególnych znaczników rejestru .P. Bez dokładnej informacji o działaniu poszczególnych znaczników warunkujących z kolei sterowanie programem miałbyś bardzo duże kłopoty w prawidłowym stosowaniu instrukcji warunkowych. Przypomnijmy zatem, że:

BCC — warunek prawdziwy, gdy znacznik przeniesienia C jest równy 0. W przeciwnym wypadku skok do danej komórki nie nastąpi.

BCS — sytuacja dokładnie odwrotna. Znacznik C musi być ustawiony, w przeciwnym wypadku skok do danej komórki nie nastąpi.

BEQ — skok zostanie wykonany tylko w wypadku, gdy ustawiony będzie znacznik N.

BNE — dokładne przeciwieństwo instrukcji BEQ. Znacznik Z musi mieć stan logiczny 0, w przeciwnym razie skok nie nastąpi.

BPL — odwrotnie niż BMI (znacznik N musi mieć stan logiczny 0).

BVC — skok zostanie wykonany gdy znacznik V—0.

BVS — odwrotnie niż przy BVC (znacznik V=1).

Na zakończenie chciałbym przedstawić prosty program dodający do siebie dwie liczby zawarte w komórkach $2733 i $2734. CASUS sumy większej od 255 pozostawiłem chwilowo (i celowo!) błędny zobaczysz znak „<"). Posiadacze Commodore 128 powinni wpisać podany program w banku „C" czyli rozpoczynając od instrukcji: A C2710 LDA #$ OD itd., w przeciwnym wypadku program nie będzie działał. Proponowałbym Czytelnikom również samodzielną analizę programu i jego działania zwłaszcza dla dość ciekawego przypadku liczb $FF i $01 (255 i 1). Dokładnie omówienie programu przedstawię już w następnej części cyklu.

(cdn)

Klaudiusz Dybowski

```
.,2710 A9 0D       LDA #$0D
.,2712 20 D2 FF    JSR $FFD2
.,2715 18          CLC
.,2716 AD 33 27    LDA $2733
.,2719 6D 34 27    ADC $2734
.,271C F0 08       BEQ $2726
.,271E B0 0C       BCS $272C
.,2720 A9 3C       LDA #$3C
.,2722 20 D2 FF    JSR $FFD2
.,2725 00          BRK
.,2726 A9 3D       LDA #$3D
.,2728 20 D2 FF    JSR $FFD2
.,272B 00          BRK
.,272C A9 3E       LDA #$3E
.,272E 20 D2 FF    JSR $FFD2
.,2731 00          BRK
```

KLAN AMSTRAD-SCHNEIDER

WERY-FIKA-TOR

W poprzednim numerze przedstawiliśmy dwa sposoby formatowania trzycalowych dysków na 210 KB. Dzisiaj kontynuujemy ten temat. Poniższy program pozwala na sprawdzenie poprawności zapisu wszystkich ścieżek na dyskietce.

Po każdym sformatowaniu dysku powinno się sprawdzić, czy wszystkie sektory istnieją i czy są poprawnie zapisane. Takie błędne formaty mogą wystąpić, gdy używamy niefirmowych lub zużytych dysków. Oczywiście można w tym celu posłużyć się programem **DISCKIT**, ale akceptuje on jedynie standardowe formaty AMSTRADA.

Nasz weryfikator (wydruk obok) umożliwia sprawdzenie dysku o dowolnym formacie. Oczywiście jest to możliwe jeśli posiadamy program zmieniający DPB (ang. Disc Parameter Block) stosownie do tego formatu. Taki DPB dla formatu 208 KB przedstawiliśmy w poprzednim numerze „Bajtka". Na uruchomienie tego, jak i innych tego typu programów komputer zwykle reaguje opuszczeniem weryfikatora. Aby wrócić do niego, należy wprowadzić **RUN 500** (RETURN) w trybie bezpośrednim.

Na ekranie w kolumnie pod literą **C** pojawią się numery ścieżek, a pod **S** numery sektorów, które program sprawdza. Poprawność lub niepoprawność jest potwierdzana stosownym komunikatem.

Jacek Kunowski

```
1 '****************************
2 '*                          *
3 '*   program WERYFIKATOR    *
4 '*   aut. JACEK KUNOWSKI    *
5 '*      Warszawa 1989       *
6 '*                          *
7 '****************************
8 '
10 MEMORY &7FFF
20 DATA &21,&23,&80,&cd,&d4,&bc,&22,&20
30 DATA &80,&79,&32,&22,&80,&1e,&00,&16
40 DATA &00,&0e,&00,&21,&30,&80,&3e,&ff
50 DATA &df,&20,&80,&00,&32,&26,&80,&C9
60 DATA &00,&00,&00,&81,&00,&00,&00
70 FOR 1=&8000 TO &8000+38
80 READ zm:POKE 1,zm
90 NEXT 1:CALL &8000
500 POKE &8027,&64:MODE 1:GOSUB 1030
510 lo=PEEK(&BE40):hi=PEEK(&BE41)
520 adpb=hi*256+lo+10
530 lo=PEEK(adpb):hi=PEEK(adpb+1)
540 adpb=hi*256+lo:PRINT#4:GOSUB 1100
550 PRINT#4,"Wgrywasz inny DPB (T/N)? ";
560 GOSUB 870:PRINT#4,a$:PRINT#4
570 IF a$="T" THEN GOSUB 910
580 PRINT#4,"Z której stacji"
590 PRINT#4," sprawdzasz dysk (A/B) ? ";
600 GOSUB 820:PRINT#4,a$:PRINT#4
610 GOSUB 1140:adpb=adpb+64*stac
630 nrsek=PEEK(adpb+15)
640 poj=PEEK(adpb+5)+1
650 lsek=PEEK(adpb+16)
660 rosek=2^(PEEK(adpb+20)+7)
670 lisc=(poj*1024)/(lsek*rosek)
680 rezsc=PEEK(adpb+13)
690 POKE &800E,stac
700 FOR j=0 TO rezsc+lisc-1
710 POKE &8010,j
720 FOR i=nrsek TO (nrsek+lsek-1)
730 POKE &8012,i
740 PRINT#2,USING "##";j;
750 PRINT#3," &";HEX$(i);
760 CALL &8000:blad=PEEK(&8026)
780 IF blad<>0 THEN GOSUB 970
                ELSE PRINT#3,"  O.K."
790 NEXT i,j
800 GOTO 1170
810 CALL 0
820 a$="":'stacja
830 WHILE NOT(a$="A" OR a$="B")
840 a$=INKEY$:a$=UPPER$(a$)
850 IF a$="A" THEN stac=0
                ELSE stac=1
860 WEND:RETURN
870 a$="":'tak czy nie
880 WHILE NOT(a$="N" OR a$="T")
890 a$=INKEY$:a$=UPPER$(a$)
900 WEND:RETURN
910 PRINT#4:a$=""
920 PRINT#4,"Z której stacji wgrywasz
                   (A,B) ? ";
930 GOSUB 820:GOSUB 1140: PRINT#4,a$
940 INPUT#4,"Nazwa pliku DPB - ",nazwa$
950 GOSUB 1100:CHAIN MERGE nazwa$
960 PRINT#4:PRINT#4,"Wgrywa się progra
m ";UPPER$(nazwa$):RETURN
970 GOSUB 1100:PRINT#3
980 IF blad=32 THEN PRINT#3,"Brak sektor
a"ELSE PRINT#3," Brak dysku"
990 PRINT#4,"  Sprawdzać dalej (T/N) ? "
";
1000 a$="":GOSUB 870
1010 PRINT#4,a$:IF a$="N" THEN 1170
1020 RETURN
1030 WINDOW #2,1,40,1,1
1040 WINDOW #3,28,40,4,25
1050 PAPER #3,3:PEN #3,1:CLS #3
1060 WINDOW #4,1,26,4,22
1070 WINDOW #5,1,40,1,3
1080 PAPER #5,1:PEN #5,0:CLS #5
1090 WINDOW #6,1,26,23,25:RETURN
1100 CLS #5:PRINT#5
1110 PRINT#5,"  POPRAWNOŚĆ FORMATU DYS
KU   C  S   ****":CLS #6:PRINT#6
1120 PRINT#6," * C-cylinder S-sektor *"
1130 RETURN
1140 IF stac=0 THEN |A
                ELSE |B
1150 GOSUB 1100:RETURN
1160 GOTO 500
1170 PRINT#4," Wychodzisz z programu";
1180 PRINT#4,:PRINT#4,SPC(17);"(T/N) ? "
;
1190 GOSUB 870:PRINT#4,a$:IF a$="N" THEN
 500
1200 FOR z=1 TO 1000:NEXT:GOTO 810
```

MicroDraft na ploterze PRN C41 firmy SONY

Opisywany program MicroDraft wraz z instrukcją otrzymaliśmy dzięki uprzejmości pana J. Cieślikowskiego z firmy InterAms. Dziękujemy.

Program Microdraft(1) jest uproszczonym system typu CAD (ang. Computer Aided Design) na 8-bitowe (*) komputery firmy Amstrad — PCW 8256/8512 i CPC 6128 (**). Rysunki sporządzone przy jego pomocy mogą być wydrukowane na drukarce mozaikowej lub narysowane na ploterze. Do pracy z drukarką służy program MPRINT, COM, natomiast z ploterem wyposażonym w interpreter języka HPGL (ang. Hewlett-Packard Graphics Language) współpracuje program MPLOT, COM. Na komputerze PCW korzystanie z plotera lub dodatkowej drukarki wymaga zastosowania interface'u CPS 8256 (2, 3). Niestety ploter PRN C41 firmy Sony (4) nie dysponuje interpreterem HPGL.

Możliwe są dwa rozwiązania tego problemu. Pierwsze z nich polega na napisaniu nowego programu MPLOT, który interpretując zbiór dyskowy produkowany przez program MDRAFT, COM, generowałby komendy plotera SONY. Jest to metoda bezpośrednia, w trakcie wykonywania której tłumaczeniu podlegają elementy rysunków MicroDraft'a takie jak: łuki, okręgi, proste itp. Drugie rozwiązanie opiera się na wykorzystaniu oryginalnego programu MPLOT, COM, tak aby wysyłał on rozkazy w języku HPGL na dysk, a nie bezpośrednio na ploter. W tym wypadku konieczne jest napisanie specjalnego programu MSONY. COM. Program ten interpretuje komendy HPGL na komendy plotera SONY.

Prostsze do wykonania okazało się rozwiązanie drugie i poniżej przedstawiamy jego opis.

SPOSÓB POSTĘPOWANIA

1 Przy pomocy programu MDRAFT. COM tworzymy na ekranie komputera rysunek. Po zakończeniu pracy rysunek zapisywany jest na dysk jako zbiór o standardowym rozszerzeniu DRW.

2. Uruchamiamy program MPLOT, COM. Jako zbiór wejściowy (Data file) podajemy nazwę zbioru utworzonego przez MDRAFT, COM. Zbiorem wyjściowym (Output file/device) jest zbiór o tej samej nazwie, ale innym rozszerzeniu — PTR.

3. Po uzyskaniu zbioru typu PTR uruchamiamy program MSONY, COM. Typowe wywołanie ma postać:

LOGICZNE I FIZYCZNE URZĄDZENIA WE/WY W SYSTEMIE CP/M PLUS

Podstawowymi urządzeniami we/wy współczesnych komputerów są monitor i klawiatura. Monitor służy do wyświetlania informacji, a klawiatura umożliwia wprowadzanie poleceń i danych do komputera.

Kolejnym typowym urządzeniem wyjściowym jest drukarka lub ploter. Są one zwykle podłączane do systemu przy pomocy złącza równoległego Centronics. Wykorzystywane jest także złącze szeregowe typu RS 232, najczęściej do pracy z modemem.

Różnorodność produkowanych monitorów, drukarek i klawiatur byłaby bardzo kłopotliwa dla użytkownika, gdyby korzystając z różnych komputerów musiałby uwzględniać szczegóły techniczne urządzeń we/wy i zmieniać swój program przy zmianie np. drukarki. Z tych powodów wprowadzono w systemie operacyjnym CP/M Plus podział na fizyczne i logicz-

M›msony nazwa_zbioru_z_rozszerzeniem_PTR ‹CR›
gdzie ‹CR› oznacza naciśnięcie klawisza RETURN.

UWAGI DOTYCZĄCE PROGRAMU MPLOT, COM

— Właściwe rozmiary rysunku na ploterze (zgodne z rzeczywistymi) można uzyskać przez odpowiednie dobranie współrzędnych pktów P1 i P2 (opcja 5),
— ploter posiada kilka piórek i kilkanaście typów linii (ciągłe, przerywane itp.). Opcja 6 — Layers pozwala określić kolor pióra i rodzaj linii, którą dana warstwa jest kreślona,
— podanie wszystkich parametrów do rysowania, a szczególnie opisów warstw może być dość długie. Dlatego warto to zrobić raz i zapamiętać te dane na dysku — opcja 9. Wczytywanie tych danych odbywa się przy pomocy opcji 8.

ZBIORY WSADOWE

Celem ułatwienia pracy z komputerem przygotowano dwa zbiory wsadowe: MP3, SUB i MP4, SUB. Pierwszy z nich pozwala na automatyczne narysowanie na ploterze Sony rysunków w formacie A3 bezpośrednio po wyjściu z programu MDRAFT, MP4, SUP spełnia te rolę dla formatu A4.
Ogólna postać wywołania zbioru wsadowego jest następująca:
M›(submit) MPx file_to_plot output_file/ /device *
gdzie
1. submit jest konieczne, jeśli nie zdefiniowaliśmy przy pomocy programu SETDEF, COM (dyskietka systemowa) możliwości wykonywania zbiorów o końcówce, SUB,
2. x=3, 4 format A3 lub A4, odpowiednio,
3. file_to_plot zbiór z MDRAFT'a. Musi mieć rozszerzenie DRW, ale nazwę podaje się bez rozszerzenia,
4. output_file/device zbiór lub urządzenie wyjściowe. Litera C oznacza ploter podłączony do złącza typu Centronics,
5. * pozwala na wykonanie programu MSONY bez zatrzymania na włożenie papieru do plotera. Czynność tę należy wykonać przed uruchomieniem zbioru wsadowego MPx.

Przykłady:
mp4 rysunek
— wykonanie rysunku A4 'RYSUNEK, DRW' na ploterze z przerwą na założenie papieru,
mp4 rysunek c *
— to samo, ale bez przerwy,
mp3 rysunek rysunek, son *
— wykonanie rysunku A3 na zbiór 'RYSUNEK, SON' bez przerwy na ustawienie papieru.
W trakcie wykonywania wsadu MPx, SUB tworzony jest tymczasowy zbiór wyjściowy XXX. PTR programu MPLOT, który jest jednocześnie zbiorem wejściowym dla programu MSONY. Ponieważ zbiór ten może być dość długi — kilkadziesiąt KB — konieczne jest zarezerwowanie odpowiednio dużego miejsca *na dyskietce (lub stacji M:).*

OPIS PROGRAMU MSONY, COM

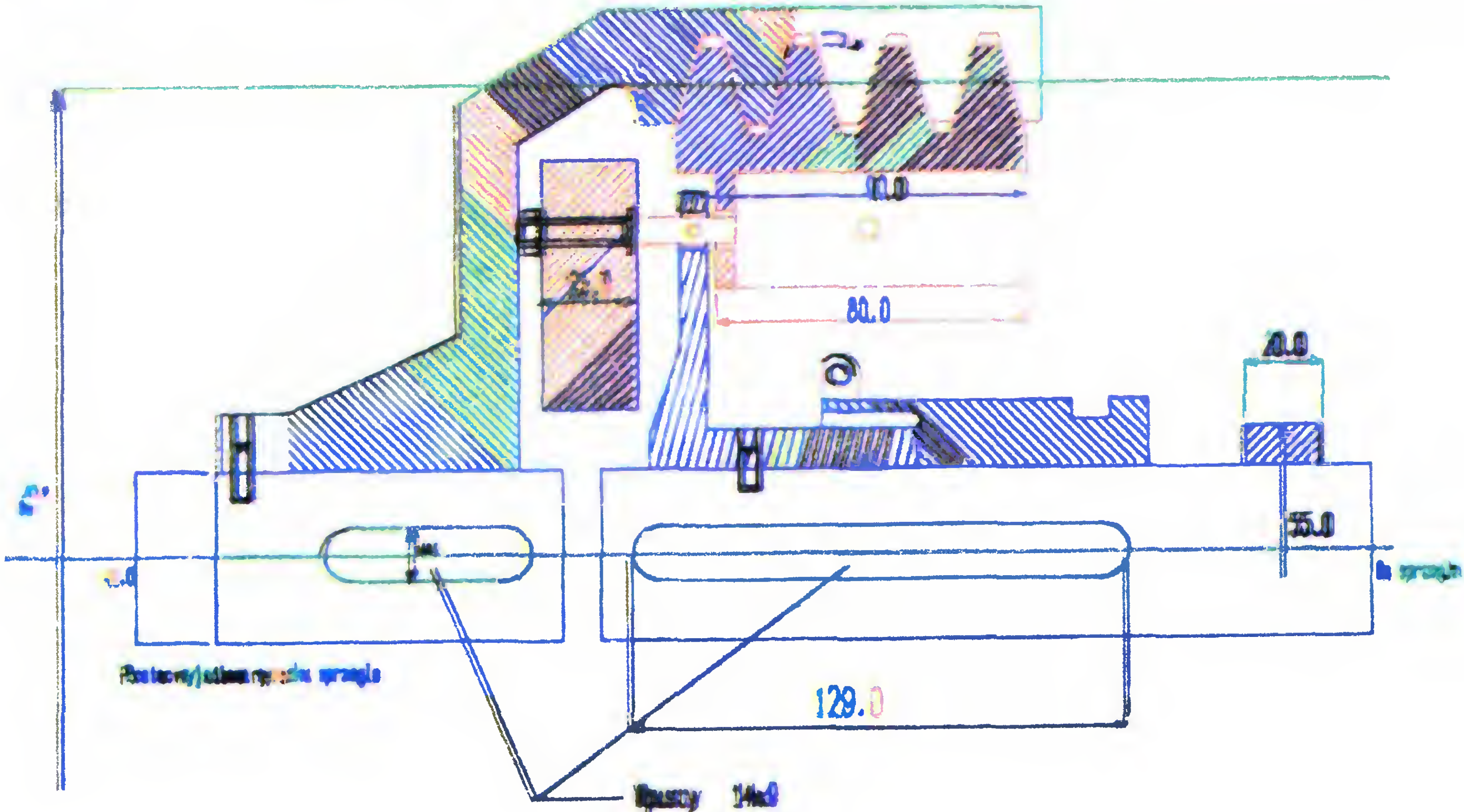
Ogólna postać wywołania programu MSONY, COM podobna jest do wywołania dla zbiorów wsadowych MPx, z tą jednak różnicą, że zbiór file_to_plot jest zbiorem otrzymanym po przetworzeniu zbioru z MDRAFT'a przez program MPLOT
M›msony file_to_plot output_file/devic *
Jeśli nie podamy żadnych parametrów przy wywołaniu, to program 'spyta się' na początku o nazwę zbioru do narysowania.
Jeśli nie podamy trzech parametrów to rozpoczęcie rysowania wymaga naciśnięcia klawisza ‹RETURN›.
Rysowanie może być przerwane w każdej chwili przez naciśnięcie dowolnego klawisza. Wznowienie pracy następuje po naciśnięciu klawisza ‹RETURN›, a ostateczne zaniechanie dalszego rysowania przez klawisz ‹EXIT›.
Program MSONY automatycznie, bez pomocy komendy systemowej DEVICE, COM, korzysta z wyjścia Centronics. Po zakończeniu urządzeniem wyjściowym jest ponownie drukarka PCW

ZBIORY NA DYSKIETCE 'DYSTRYBUCYJNEJ'

SUBMIT, COM
MSONY, COM MP3,SUB MP4,SUB
MPLOT, COM SET3,PLT SET4,PLT
TEXT.FON
Zbiory SETx, PLT zawierają dane do programu MPLOT, COM. Podano odpowiednie współrzędne pktów P1 i P2, oraz zdefiniowano kolory i rodzaj linii dla kilku warstw:
Layer 0, 8, 16, 24 — linie ciągłe, kolory 1, 2, 3, 4
(black, blue, green, red)
Layer 3, 11, 19, 27 — linie przerywane, kolory 1, 2, 3, 4
odpowiednio
Zbiorem wejściowym jest XXX, DRW, a wyjściowym XXX, PTR.

PODSUMOWANIE

Program MSONY pozwala na korzystanie w wygodny sposób z plotera SONY do sporządzania rysunków uzyskanych z programu MDRAFT. Ploter umożliwia przynajmniej ponad dwukrotne przyśpieszenie pracy. Wykonanie programów MPLOT i MSONY na dysk jest bardzo szybkie — pojedyncze minuty. Dodanie programu typu 'spooler' drukarki pozwoliłoby na późniejsze rysowanie na ploterze w 'tle' przy współbieżnej pracy na komputerze innego programu.
Zestaw PCW 8256/8512 z ploterem PRN C41 firmy Sony pracujący z programem MicroDraft stanowi atrakcyjną alternatywę dla osób prywatnych lub instytucji nie dysponujących większymi zasobami finansowymi na zakup sprzętu firm IBM i ROLAND.
Ilustracją działania programu MSONY jest rysunek z (1) narysowany na ploterze SONY.



(*) Istnieje wersja programu Micro Draft na komputer IBM PC. Dzięki zgodności zapisu rysunków na dysku z wersją 8-bitową istnieje możliwość wymiany rysunków między tymi komputerami. Jest to szczególnie ułatwione przy korzystaniu z dodatkowej 5-calowej stacji dysków na PCW.
(**) Praca na CPC 6128 jest istotnie spowolniona z powodu braku ramdysku. Mała pojemność dyskietki jest także przyczyną częstych zmian stron dyskietki, ponieważ Microdraft zajmuje więcej niż 180 KB.

Jonasz Mayer

Literatura:
1. J. Wichniewicz, 'Komputer' 12/87, str. 39
2. J. Mayer, Bajtek 6/88, str. 14
3. J. Mayer, „Test interface'u SCA do PCW", 'Bajtek' 3/89
4. J. Mayer, „Test plotera PRN C41 firmy SONY" 'Bajtek' 3/89

---

ne urządzenie we/wy, pozostawiając producentowi sprzętu problemy ich właściwej implementacji i wzajemnego przypisania. *Urządzenia logiczne są dla* programisty lub użytkownika takie same na każdym komputerze pracującym w tym systemie. Natomiast dostępne urządzenia fizyczne są zależne od oferowanej przez wytwórcę konfiguracji.

URZĄDZENIA LOGICZNE

W systemie CP/M Plus mamy 5 logicznych urządzeń we/wy:
— CONIN: (Console Input)
— CONOUT: (Console Output)
— AUXIN: (Auxiliary Input)
— AUXOUT: (Auxiliary Output)
— LST: (List Device)
Pierwsze dwa służą do wprowadzania danych i wyświetlania wyników i są zwykle przypisane do klawiatury *i monitora. Urządzenia typu AUX realizują* dodatkowy kanał informacji i typowo są wykorzystane do obsługi złącza RS 232C, LST odpowiada drukarce i najczęściej obsługuje złącze równoległe.
Komenda systemowa PIP pozwala na traktowanie tych urządzeń w sposób podobny do zbiorów dyskowych. W szczególności:
A›PIP zbior=CONIN:
pozwala na utworzenie zbioru dyskowego przy pomocy klawiatury, a
A›PIP LST: = zbior
umożliwia wysłanie tego zbioru na drukarkę.
W Turbo Pascalu pięciu logicznym urządzeniom we/wy odpowiadają trzy pliki:
CON: (Console)
AUX: (Auxiliary)
LST: (List Device)
Zasady korzystania z tych plików są prawie takie same jak dla zwykłych plików dyskowych. Poniższa procedura wyprowadzi na drukarkę zmienną tekstową napis,

```
procedure pisanie_po_pliku_wy;
var f: text;
  napis: string (,255,)
begin
  napis: = 'xxxxxxxxx'
  assign (f, 'LST:'); (* drukarka *)
  rewrite (f);
  write (f, napis);
  close (f);
end; (* pisanie po pliku *)
```

Następna procedura przeczyta na zmienną napis tekst z pliku AUX,

```
procedure czytanie_pliku_we;
var f: text;
  napis: string (,255,);
begin
  assign (f, 'AUX:');
  reset (f);
  read (f, napis);
  close (f);
end; (* czytanie pliku we *)
```

W Turbo Pascalu oprócz tych urządzeń we/wy mamy jeszcze inne:
INPUT
OUTPUT
KBD
LST
Ich obsługa jest prostsza, a w szczególności nie jest konieczne ich otwieranie i zamykanie.
READ (INPUT, napis) odpowiada instrukcji READ (napis) i pozwala wczytać z klawiatury zmienną tekstową napis.
WRITE (OUTPUT, napis) odpowiada instrukcji WRITE (napis) i wypisuje na monitorze zmienną napis.
Urządzenie KBD jest stosowane przy czytaniu pojedynczych klawiszy bez pośrednictwa bufora
Kolejna procedura oczekuje w pętli na przyciśnięcie klawisza i zwraca odpowiadający mu znak w zmiennej ch,

```
procedure Key (ch: char);
begin
  repeat until KeyPressed;
  read (KBD, ch);
end; (* KEY *)
```

Najprostszym sposobem wysłania tekstu na drukarkę jest instrukcja:
writeln (LST, 'TEKST NA DRUKARKE');
Wcześniejsza metoda pisania po drukarce jest bardziej skomplikowana, ale dzięki zgodności z obsługą typowego pliku dyskowego pozwala na jednolite traktowanie każdego pliku. Jest to wygodne, jeśli o urządzeniu wyjściowym będzie miał decydować użytkownik programu.

FIZYCZNE URZĄDZENIA WE/WY.

W 8-bitowych Amstradach mamy zawsze przynajmniej dwa fizyczne urządzenia we/wy:
CRT (Cathode Ray Tube)
LIP (Line Printer)
CRT obejmuje klawiaturę i monitor, a LPT jest drukarką. *W przypadku PCW jego własna drukarka nie jest* obsługiwana przez złącze Centronics. Dodanie interface'u typu CPS 8256 wprowadza do systemu dwa kolejne urządzenia fizyczne:
CEN (Centronics)
SIO (Serial Input/Output)
CEN pozwala na podłączenie standartowej drukarki lub plotera poprzez złącze równoległe, a SIO realizując złącze szeregowe typu RS 232C umożliwia korzystanie m.in. z modemu lub innych urządzeń wyposażonych w RS'a.
Do Amstrada CPC 6128 (464/664) można także podłączyć interface typu RS 232C. Ze względu na boga-



te oprogramowanie komunikacyjne najlepiej nadaje się do tego produkt angielskiej firmy Pace. Po dołączeniu interface'u Amstrad CPC dysponuje trzecim fizycznym urządzeniem we/wy:

SIO (Serial Input/Output)

Standardowo po włączeniu komputera mamy następujące odwzorowanie między urządzeniami fizycznymi i logicznymi:

urządzenie fizyczne	urządzenie logiczne
CRT	CON: (CONIN:, CONOUT:)
LPT	LIST:
SIO	AUX: (AUXIN:, AUXOUT:)
CEN	—

KOMENDA DEVICE

Do zmiany wzajemnego przypisania urządzeń logicznych i fizycznych służy polecenie systemowe DEVICE (zbiór DEVICE, COM). Przykładowo:

A>DEVICE AUXOUT: = CEN

przypisze logicznemu urządzeniu AUXOUT: jego fizyczny odpowiednik — złącze równoległe Centronics, a

A>DEVICE LST: = NULL

przyporządkuje urządzeniu logicznemu LST: urządzenie puste NULL. Spowoduje to pominięcie wszelkich wydruków, będą one wysyłane na nieistniejące urządzenie fizyczne.

TABLICA PARAMETRÓW SYSTEMOWYCH

W CP/M'ie istotne informacje systemowe przechowywane są w specjalnej, 100-elementowej tablicy SCB (ang. System Control Block), której adres można uzyskać przy pomocy następującej funkcji pascalowej:

```
Function SCBaddr: integer;
var b: integer;
begin
   b: = $003A;
   SCBaddr: = BDOSHL (49, addr (B));
end; (* SCBaddr *)
```

W Amstradzie adres ten wynosi FB9C H, 10 bajtów tej tablicy określa 5 wektorów przypisań po jednym, na każde urządzenie logiczne:

adres wektora	urządzenie
22 H	CONIN:
24 H	CONOUT:
26 H	AUXIN:
28 H	AUXOUT:
2A H	LST:

Każdy bit wektora odpowiada innemu urządzeniu fizycznemu. W szczególności mamy następujące przyporządkowanie:

urządzenie fizyczne	wartość
CEN	10 H
SIO	20 H
LPT	40 H
CRT	80 H

Wektory te dostępne przez program użytkownika mogą być w nim zmieniane. Przykładowo:

MEM (,$FB9C+$2B,): = $80;

spowoduje w programie pascalowym przypisanie logicznego urządzenia LST: do fizycznego urządzenia jakim jest monitor. Wydruki zamiast na drukarkę powędrują na ekran.

Deklaracje:

```
const
  Cen = $10; Sio: = $20; LPT = $40; CRT =$80;
procedure AssignLST (a: byte);
begin
  MEM (,$FB9C+2B,): = a;
end; (* AssignLST *)
```

ułatwią korzystanie z tych możliwości we własnych programach pisanych w Turbo Pascalu.

Wywołanie

AssignLST (CEN);

pozwala przypisać logicznemu urządzeniu LST: fizyczne złącze Centronics.

KORZYSTANIE Z DODATKOWEJ DRUKARKI (LUB PLOTERA) NA PCW.

Jeśli chcemy mieć możliwość jednoczesnego korzystania z oryginalnej drukarki PCW i drukarki dodatkowej, a nie używamy złącza szeregowego, to bardzo wygodnym rozwiązaniem jest następujące przypisanie

DEVICE AUXOUT: = CEN

Komenda

PIP LST: = zbiór, txt

wydrukuje zbiór dyskowy na oryginalnej drukarce PCW, natomiast polecenie PIP AUX: = zbiór, txt prześle go na drugą drukarkę.

W Turbo Pascalu problem wygląda podobnie:

writeln (lst, 'TEXT');

wyśle na oryginalną drukarkę napis 'TEXT', a

writeln (aux, 'TEXT');

wydrukuje ten sam napis na drugiej drukarce.

Jonasz Mayer

DEKODER TRANSMISJI RADIOWEJ

Z końcem 1985 roku Rozgłośnia Harcerska Polskiego Radia rozpoczęła emitowanie programów komputerowych. Audycja „Radiokomputer" powstała na życzenie słuchaczy i do dziś prezentuje programy przez nich opracowywane i nadsyłane.

Chociaż pierwsze z emitowanych programów przeznaczone były dla Spectrum, nie stały się popularne. Stosowany w Spectrum standard zapisu informacji na taśmie magnetofonowej jest niezwykle niepodatny do emisji, wrażliwy na zakłócenia. Przez to tylko nieznaczny procent programów dawał się potem odtworzyć i używać.

Sytuacja poprawiła się, gdy p. Tomasz Jordan, założyciel i prowadzący „Radiokomputer", rozpoczął emisję programów na Atari i Commodore. W przypadku Atari ma miejsce stuprocentowa pewność co do jakości nagranych programów, zaś Commodore wczytuje ok. 70%.

Spectrumowcy czuli się nadal zawiedzeni. Liczne próby coraz to nowych systemów nie dawały rezultatów i „Radiokomputer" stracił sporą liczbę słuchaczy.

Teraz nareszcie doczekali się oni pewnego systemu transmisji. Firma „SZOK" ze Świebodzina, w osobie Mirosława Bereźnickiego i Zygmunta Rozynka, opracowała ostateczną wersję programu transmisji, o nazwie „Dekoder Transmisji Radiowej PLUS", w skrócie DTR+. Warto nadmienić, iż „SZOK" zajmuje się tworzeniem oprogramowania dla Atari, wykorzystując przy tym Spectrum jako urządzenie pomiarowe. Przy tak wielkiej znajomości Spectrum firma ta, jako trzecia z rzędu, podjęła się i — co najważniejsze — stworzyła niezawodny program DTR+.

Program przeznaczony do emisji na falach radiowych jest kodowany przez sam „Radiokomputer", a następnie emitowany. Po nagraniu go na taśmę z radia, należy poddać go odkodowaniu. Zrobi to za Was zamieszczony obok program „DTR+", który trzeba wpisać (najpierw procedura ładująca — linie 100—120 z pierwszego wydruku) i nagrać przez SAVE „DTR+"LINE 1. Następnie, po wpisaniu drugiej części, należy nagrać blok kodu za poprzednio nagraną procedurą ładującą.

Tak przygotowany program DTR+ gotowy jest do użytku. Po wczytaniu uruchamia się ona automatycznie i pojawia się mini-menu, dzięki któremu możliwe jest wczytanie zakodowanego programu (LOAD), nagranie odkodowanej wersji na taśmę (SAVE) oraz powrót do systemu (RESET).

Emitowany w radio program zapisany jest w postaci jednego bloku, około dwukrotnie dłuższego niż cały program źródłowy. Program DTR+ odkodowuje go i nagrywa w kilku blokach tak, że jest gotowy do użytku. Dodatkową zaletą jest to, że po odkodowaniu użytkownik dysponuje wersją nagraną na specyficznym dla jego magnetofonu ustawieniu głowicy.

Muszę zmartwić graczy: „Radiokomputer" nie emituje gier! Nie będziecie więc mieli czego tak szukać, chyba, że interesują Was również programy użytkowe, bo tylko takie można tam znaleźć. Ich autorami są w większości słuchacze „Radiokomputera", choć zdarza się emisja programów zachodnich, np. „SuperCode".

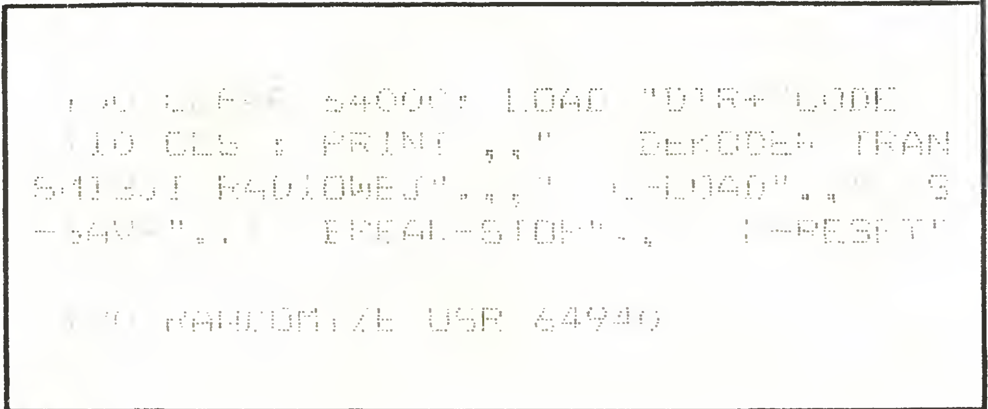
Wszystkich zainteresowanych współpracą z „Radiokomputerem" prosimy o nadsyłanie listów i programów na adres:

Rozgłośnia Harcerska
00-950 Warszawa
ul. M. Konopnickiej 6,
z dopiskiem „Radiokomputer"

Istnieje jedno ograniczenie dla programów przeznaczonych do zakodowania i emisji: wszystkie bloki muszą posiadać nagłówki.

„Radiokomputer" emitowany jest na falach ultrakrótkich i średnich, w każdy czwartek w godz. 16.00—16.20 oraz wtorek — godz 22.50—24.00. Blok emisji programów to godz. 23.25—24.00

Marcin Przasnyski



```
100 CLEAR 64000: LOAD "DTR+"CODE
110 CLS : PRINT ,,"  DEKODER TRAN
SMISJI RADIOWEJ",,,"  L-LOAD",,"  S
-SAVE",,"  E-REAL-STOP",,"  R-RESET"

120 RANDOMIZE USR 64940
```

Procedura ładująca

```
 150 CLEAR 64000: LET s=0: FOR a=64
940 TO 65534: READ b: POKE a,b: LET s=s+
b: NEXT a
 160 IF s=69549 THEN  SAVE "DTR+"CODE 64
940,595: STOP
 170 PRINT "Popraw dane": STOP
 200 DATA 1,16,13,205,10,32,17,196,253,2
05,55,254,62,16,50,71,80,62,8,50,39,87,2
4,48,62
 210 DATA 83,90,79,75,60,13,13,32,32,32,
32,32,32,32,83,119,105,101,98,111,100,12
2,105,110,32
 220 DATA 32,49,57,56,56,114,174,194,9,2
04,9,219,9,237,9,0,0,0,0,0,0,0,253,54,56

 230 DATA 0,253,203,48,222,49,229,253,62
,56,50,130,88,50,98.88,205,91,254.245,20
5,65,254,241,254
 240 DATA 82,202,0,0.254,76,40,78,254,83
,204,156,255,24,221,1,21,0,205,10,32,24,
13,203,39
 250 DATA 60,33,228,253,35,61,32,252,94,
35,86,62,32,215,26,19,203,127,40,249,203
,191,215,201,1
 260 DATA 21,0,17,32,0,24,6,1,8,0,17,159
,1,205,10,32,62,32,215,27,122,179,32,248
,201
 270 DATA 33,8,92,54,0,126,254,0,40,251.
201,205,73,254,253,54,56,0,62,248,50,98,
88,243,17
 280 DATA 5,0,33,239,253,14,101,205,231,
254,33,69,95,237,91,241,253,25,218,144,2
55,33,248,92,14
 290 DATA 86,205.231,254,1,8,0,205,10,32
,253,54,57,13,33,248,92,34,237,253,229,2
37,91,239,253
 300 DATA 167,237,82,40,49,257,53,57,202
,144,255,225,229,126,254,4,210,144,255,2
05,39,254,225,6,10
 310 DATA 35,126,254,32,56,4,254,128,56,
2,62,32,215,16,241,62,13,215,35,94,35,86
,1,5,0
 320 DATA 9,25,24,194,253,53,56,251,62,
,211,254,195,252.253,217,49.227,253,22,1
1,46,50,62,25
 330 DATA 38.1,205,76,253,48,243.45,32,2
44,62,22.22,13,205,93,255,217,185,217,32
,228.22,14,62
 340 DATA 19,24,2,62,17,205,93,255,217,1
19,169,79,35,27,122,179,217,32,240,217,1
75,185,32,110,201
 350 DATA 71,16,254,203,18,63,203,26,62,
8,203,122,32,1,122,211.254.62,127,219,25
4,31,48,94,1
 360 DATA 64,37,219,254,161,187,32,4,16,
248,24,67,95,201,38,3,205,35,255,62,26,3
7,32,248,96
 370 DATA 205,35,255,120,132,254,44,201,
205,74,255,48,41,62,25,205,84,255,56,249
,62,25,205,84,255
 380 DATA 111,62,26,205,84,255,189,38,3,
48,1,37,62,24,46,128,205,76,255,203,29,6
2,25,38,3
 390 DATA 48,245,125,201,203,66,194.232,
254,205,73,254,17,37,21,205,31,254,195,2
23,254,253,203,56,126

 400 DATA 202,252,253,237,115,61,92,62,2
48.50,130,88,17,162,9,205,31,254,205,91,
254,205,65,254,33
 410 DATA 248,92,34,237,253,237,91,239,2
53,167,237,82,202,252,253,221,42,237,253
,205,246,255,17,17,0
 420 DATA 62,0,205,194,4,221,42,237,253,
205,246,255,221,94,11,221,86,12,213,1,17
,0,221,9,221
 430 DATA 229,62,255,205,194,4,225,209,2
3,24,197,17,253,255,122,179,200,27.24,23
0
```

KUBA ROZPRUWACZ

Dziś rano uciekł z więzienia znany przestępca i włamywacz, Kuba Rozpruwacz. Jesteś jedyną osobą, która widziała go zaraz po ucieczce. Dysponujesz starą maszyną do sporządzania portretów pamięciowych. Ma ona zakodowane po kilka cech każdej części twarzy. Są to: kształt głowy, gęstość włosów, wielkość oczu, gęstość brwi, wielkość nosa, uszu, ust i wąsów. Kuba nie nosi brody, gdyż go łaskocze. Nie przylepia też sztucznej z powodu alergii na klej.

Każdą cechę możesz ustalić tylko raz. Po podaniu ostatniej, maszyna tworzy gotowy portret. Twoje zadanie ogranicza się więc do zmiany wielkości kolejnych części twarzy.

Przedstawiony poniżej program symuluje taką właśnie maszynę. Poszczególne cechy generowane są losowo. W sumie możliwe jest stworzenie 3888 różnych twarzy. Portret rozpoczynany jest od głowy. Można ją zwiększać i zmniejszać klawiszami 1 i 2. Klawisz 0 służy do przejścia do następnej cechy.

Po ustaleniu wielkości wąsów, twój portret porównywany jest z oryginałem. Dowiadujesz się, które części podałeś źle, oraz jaka jest odchyłka portretu od oryginału. I to już koniec. Możesz grać od początku, tworzony jest nowy Kuba i ty musisz stworzyć jego portret.

Program powstał na pomyśle gry „PHOTOFIT". Było to tworzenie portretów przez podawanie określających ich przymiotników, np. dłuższy, węższy, grubszy, oczywiście po angielsku.

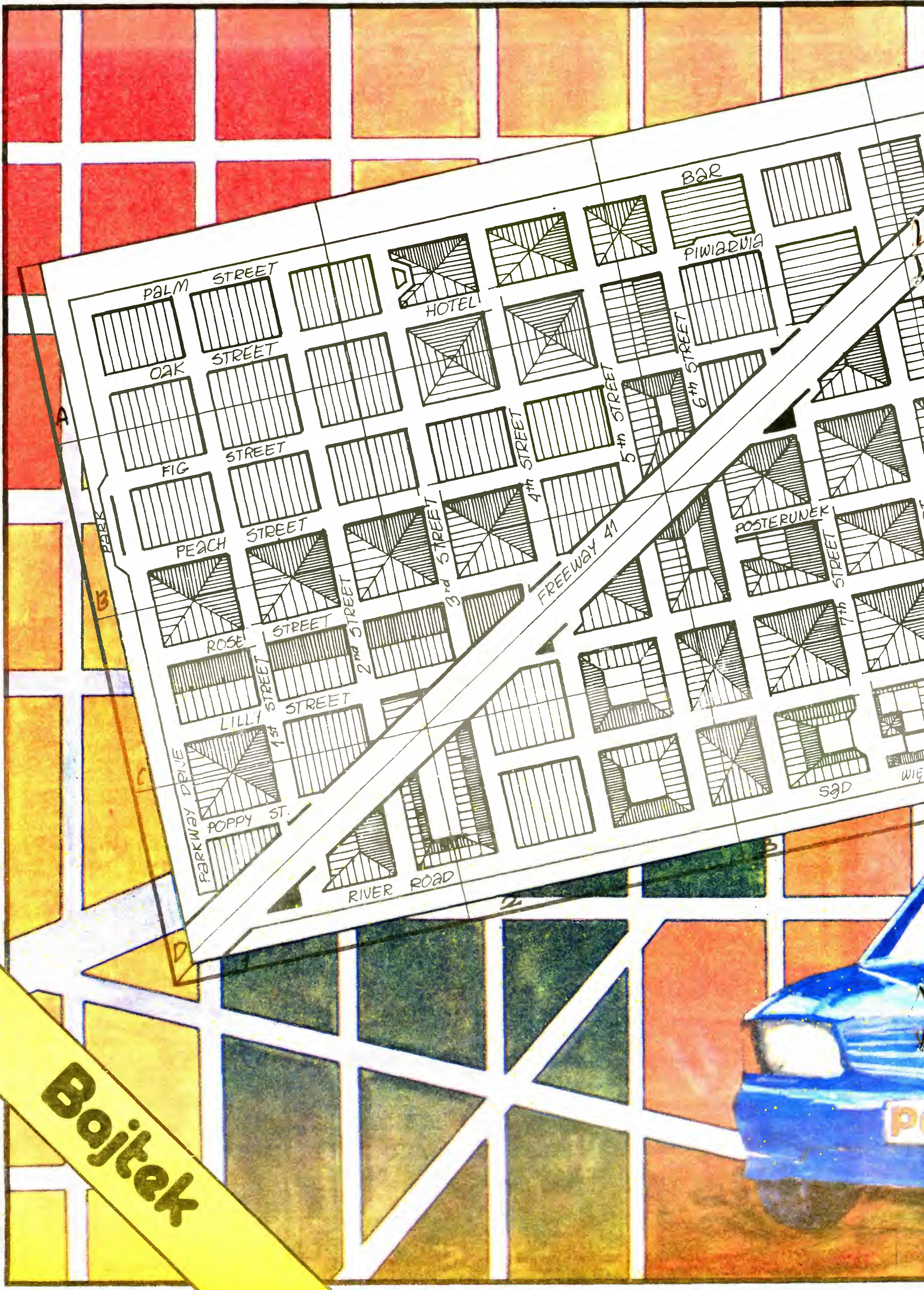
Ja ograniczyłem się tylko do wielkości, różnie rozumianej.

Proponuję rozbudować program według własnego uznania. Jego główną częścią są podprogramy 5000—5700, gdzie tworzone są części twarzy na podstawie zmiennych zawartych w tablicy a. Podprogram 6000 pozwala na zmianę wielkości odpowiedniej części. Zmiennymi wejściowymi są: wielkość początkowa, wielkość najmniejsza i największa oraz podprogram rysujący daną część.

Program został „potraktowany" weryfikatorem z Bajtka 6/88, w celu uniknięcia pomyłek przy wprowadzaniu listingu. Nie należy więc wpisywać instrukcji REM na końcu linii, lecz po wpisaniu programu wykonać RANDOMIZE USR 23380. Oczywiście, w pamięci musi znajdować się weryfikator. Potem wystarczy porównać wartości przy REM w programie i w listingu.

Marcin Przasnyski

```
   0>REM KUBA ROZPRUWACZ: REM 69
   5 DIM a(8): DIM o(8): REM 3B
  10 OVER 0: INK 0: PAPER 7: BORDER 7: CLS : REM A3
  20 LET gl=INT (RND*3)+1: REM BB
  30 LET wl=INT (RND*4): REM 64
  40 LET oo=INT (RND*2)+1: REM D8
  50 LET br=INT (RND*3): REM 67
  60 LET no=INT (RND*3)+1: REM ED
  70 LET uy=INT (RND*2)+1: REM 06
  80 LET us=INT (RND*3)+1: REM 0C
  90 LET wa=INT (RND*3): REM 93
 100 LET a(1)=gl: LET a(2)=wl: LET a(3)=oo: LET a(4)=br: LET a(5)=no: LET a(6)=uy: LET a(7)=us: LET a(8)=wa: REM A1
 110 GO SUB 2000: REM 12
 120 FOR f=1 TO 8: LET a(f)=5: NEXT f: REM E3
 200 FOR f=50 TO 110 STEP 10: REM 3E
 210 FOR g=0 TO 4: REM C8
 220 PLOT INK 1;f+g,80: DRAW INK 1;0,-80: REM 2E
 230 NEXT g: REM 47
 240 NEXT f: REM 50
 250 PLOT INK 1;50,0: DRAW INK 1;60,0: PLOT INK 1;50,80: DRAW INK 1;60,0: REM 27
 260 PLOT 65,80: DRAW 7,12: PLOT 95,80: DRAW -7,12: REM 76
 300 PRINT AT 1,19;"To jest";AT 3,16;"KUBA ROZPRUWACZ": REM E7
 310 PRINT AT 5,15;"Uciekl dzis rano";AT 6,15;"z wiezienia.": REM 8C
 320 PRINT AT 7,15;" Zapamietaj jego";AT 8,15;"twarz. Bedziesz";AT 9,15;"ja musial potem";AT 10,15;"odtworzyc.": REM D3
 330 PRINT AT 14,15;"nacisnij ENTER": REM B1
 340 IF INKEY$<>CHR$ 13 THEN GO TO 340: REM CC
 350 CLS : REM 60
 400 FOR f=50 TO 110 STEP 10: REM 07
 410 FOR g=0 TO 4: REM 91
 420 PLOT INK 1;f+g,80: DRAW INK 1;0,-80: REM F7
 430 NEXT g: REM 10
 440 NEXT f: REM 19
 450 PLOT INK 1;50,0: DRAW INK 1;60,0: PLOT INK 1;50,80: DRAW INK 1;60,0: REM F0
 460 PLOT 65,80: DRAW 7,12: PLOT 95,80: DRAW -7,12: REM 3E
 500 PRINT AT 1,16;"KUBA ROZPRUWACZ": REM 99
 505 PRINT #0;"1-mniejsze, 2-wieksze,0-nastepne": REM 57
 510 PRINT AT 5,16;"Najpierw GLOWA": REM 1A
 520 LET n=1: LET a(n)=1: REM CB
 530 LET sub=5000: REM 0E
 540 LET min=1: LET max=3: REM E0
 550 GO SUB 6000: REM 80
 555 PRINT AT 5,16;"Teraz WLOSY",: REM 5C
 560 LET n=2: LET a(n)=2: REM F7
 570 LET sub=5100: REM 9B
 580 LET min=0: LET max=3: REM 06
 590 GO SUB 6000: REM A8
 595 PRINT AT 5,16;"Potem OCZY",: REM 29
 600 LET n=3: LET a(n)=1: REM 1F
 610 LET sub=5200: REM 29
 620 LET min=1: LET max=2: REM 2E
 630 GO SUB 6000: REM D0
 635 PRINT AT 5,16;"I BRWI",: REM 80
 640 LET n=4: LET a(n)=1: REM 49
 650 LET sub=5300: REM B6
 660 LET min=0: LET max=2: REM 54
 670 GO SUB 6000: REM F8
 675 PRINT AT 5,16;"Oraz NOS",: REM 69
 680 LET n=5: LET a(n)=1: REM 73
 690 LET sub=5400: REM 44
 700 LET min=1: LET max=3: REM 80
 710 GO SUB 6000: REM 20
 715 PRINT AT 5,16;"Teraz USZY",: REM B8
 720 LET n=6: LET a(n)=1: REM 9D
 730 LET sub=5500: REM D1
 740 LET min=1: LET max=2: REM A6
 750 GO SUB 6000: REM 48
 755 PRINT AT 5,16;"Potem USTA",: REM C1
 760 LET n=7: LET a(n)=1: REM C7
 770 LET sub=5600: REM 5F
 780 LET min=1: LET max=3: REM D1
 790 GO SUB 6000: REM 71
 795 PRINT AT 5,16;"Wreszcie WASY",: REM 3B
 800 LET n=8: LET a(n)=2: REM F4
 810 LET min=0: LET max=2: REM EB
 820 LET sub=5700: REM F7
 830 GO SUB 6000: REM 99
 900 PRINT AT 5,16;"
": REM 67
 910 INPUT 1: REM CC
 920 PRINT #0;"Zaraz sie przekonamy ...": REM 1C
 930 LET o(1)=a(1)-gl: LET o(2)=a(2)-wl: LET o(3)=a(3)-oo: LET o(4)=a(4)-br: LET o(5)=a(5)-no: LET o(6)=a(6)-uy: LET o(7)=a(7)-us: LET o(8)=a(8)-wa: REM DA
 940 LET su=0: FOR f=1 TO 8: LET su=su+ABS o(f): NEXT f: REM 31
 950 IF su=0 THEN PRINT BRIGHT 1;AT 5,16;"GRATULACJE !!!";AT 8,17;"TO ON !": GO TO 1000: REM 03
 960 PRINT AT 5,16;"Niestety ...";AT 7,16;"Kuba wyglada";AT 8,16;"troche inaczej.": REM 51
 965 PRINT AT 11,16;"Roznia sie:": REM 48
 966 OVER 1: REM F4
 967 IF o(1)<>0 THEN PRINT TAB 17;"glowa": REM 83
 968 IF o(2)<>0 THEN PRINT TAB 17;"wlosy": REM AA
 969 IF o(3)<>0 THEN PRINT TAB 17;"oczy": REM 33
 970 IF o(4)<>0 THEN PRINT TAB 17;"brwi": REM 25
 971 IF o(5)<>0 THEN PRINT TAB 17;"nos": REM C3
 972 IF o(6)<>0 THEN PRINT TAB 17;"uszy": REM 52
 973 IF o(7)<>0 THEN PRINT TAB 17;"usta": REM 37
 974 IF o(8)<>0 THEN PRINT TAB 17;"wasy": REM 41
 999 PRINT AT 10,16;"Odchylka = ";su: REM 2C
1000 INPUT "Jeszcze raz ? ";a$: REM C0
1005 LET a$=a$+"t": LET a$=a$(1): REM 2D
1010 IF a$="t" OR a$="T" THEN RUN : REM 5F
1999 STOP : REM BE
2000 REM Kuba: REM 4E
2010 GO SUB 5000: REM glowa: REM 91
2030 GO SUB 5100: REM wlosy: REM 2E
2040 GO SUB 5200: REM oczy: REM 24
2050 GO SUB 5300: REM brwi: REM 83
2060 GO SUB 5400: REM nos: REM 8E
2070 GO SUB 5500: REM uszy: REM 89
2080 GO SUB 5600: REM usta: REM DA
2090 GO SUB 5700: REM wasy: REM 51
2100 RETURN : REM 40
4999 STOP : REM 82
5000 PLOT 80,90: DRAW 0,80,PI/(2-a(1)/3): DRAW 0,-80,PI/(2-a(1)/3): REM 15
5010 RETURN : REM A9
5100 IF a(2)=0 THEN RETURN : REM 4B
5110 FOR f=-15 TO 15 STEP 8-2*a(2): PLOT f+80,170-f*f/18*(3-a(1)): DRAW 0,5: NEXT f: REM F6
5120 RETURN : REM 18
5200 IF a(3)=1 THEN PLOT 63,140: DRAW 10,0,2*PI/3: DRAW -9,0,2*PI/3: PLOT 97,140: DRAW -10,0,2*PI/3: DRAW 9,0,2*PI/3: REM 11
5210 IF a(3)=2 THEN CIRCLE 71,140,5: CIRCLE 89,140,5: REM 15
5220 RETURN : REM 7C
5300 IF a(4)=0 THEN RETURN : REM 18
5310 FOR f=64 TO 72 STEP 6-2*a(4): PLOT f,146: DRAW 0,2: NEXT f: REM 12
5320 FOR f=86 TO 96 STEP 6-2*a(4): PLOT f,146: DRAW 0,2: NEXT f: REM 54
5330 RETURN : REM EA
5400 PLOT 80,138: REM 61
5410 DRAW -2-2*a(5),-19: DRAW 4+4*a(5),0: DRAW -2-2*a(5),19: REM F8
5420 PLOT 80-3/2*a(5),121: PLOT 80+3/2*a(5),121: REM ED
5430 RETURN : REM 4F
5500 PLOT 80-19/(1-a(1)/6),122: REM E4
5510 DRAW -4-2*a(6),20: DRAW 1+2*a(6),2*a(6): DRAW 3,-2: REM 6C
5520 PLOT 80+19/(1-a(1)/6),122: REM F6
5530 DRAW 4+2*a(6),20: DRAW -1-2*a(6),2*a(6): DRAW -3,-2: REM AE
5540 RETURN : REM BD
5600 IF a(7)=3 THEN PLOT 64,110: DRAW 32,0,PI/4: REM E4
5610 IF a(7)=2 THEN PLOT 68,110: DRAW 24,0,PI/4: REM ED
5620 IF a(7)=1 THEN PLOT 72,110: DRAW 16,0,PI/4: REM ED
5630 RETURN : REM 17
5700 IF a(8)=0 THEN RETURN : REM B2
5710 FOR f=75 TO 85 STEP 5-2*a(8): PLOT f,117: DRAW 0,-3: NEXT f: REM D9
5720 RETURN : REM 72
6000 IF INKEY$<>"" THEN GO TO 6000: REM 5C
6002 GO SUB sub: REM C9
6005 IF INKEY$="1" AND a(n)>min THEN OVER 1: GO SUB sub: OVER 0: LET a(n)=a(n)-1: GO SUB sub: REM BF
6010 IF INKEY$="2" AND a(n)<max THEN OVER 1: GO SUB sub: OVER 0: LET a(n)=a(n)+1: GO SUB sub: REM C3
6020 IF INKEY$="0" THEN RETURN : REM C2
6040 GO TO 6005: REM 10
```



BAR
PIWIARNIA
HOTEL
PALM STREET
OAK STREET
FIG STREET
PEACH STREET
ROSE STREET
LILLY STREET
POPPY ST.
RIVER ROAD
PARKWAY DRIVE
PARK
FREEWAY 41
1st STREET
2nd STREET
3rd STREET
4th STREET
5th STREET
6th STREET
7th STREET
POSTERUNEK
SAD
Bajtek

POLICE QUEST

Firma Sierra On-Line nie dała długo czekać na następną grę. Już w pierwszych miesiącach 1987 roku pojawiła się "POLICE QUEST" — jak dotąd najbardziej rozbudowana i najtrudniejsza gra Sierry.

Miasteczko Lytton to niewielka, schematyczna miejscowość. Przecinające się pod kątem prostym równoległe ulice są prawie identyczne. Niewielki hotel, park, piwiarnia, bar i klub policyjny dopełniają tego małomiasteczkowego obrazu.

Miejscowy posterunek też nie jest niczym nadzwyczajnym. Skąpa drogówka i jeszcze uboższa brygada antynarkotyczna nudzą się całymi dniami. Ty, czyli policjant Sonny Bonds, nie masz nic ciekawego do roboty. Codzienny patrol wąskich ulic Lytton starym Fordem jest czynnością tak rutynową, że już niemal zasypiasz.

Ten dzien na pozór nie różnił się od innych. Poranny prysznic, potem nudzenie starego Dooleya na odprawie — wszystko zwyczajnie. Podbudowuje Cię myśl, że do emerytury już tylko 3 lata. Nuuudy.

Ale już przed posterunkiem niespodzianka: przebita opona w wozie. Szybko zmieniasz koło i postanawiasz na przyszłość dobrze lustrować wóz przed wyruszeniem na trasę.

Podczas patrolu nagle odzywa się radiotelefon: wypadek na rogu Fig i Czwartej! Szybko podążasz na skrzyżowanie.

A tam... prawie nowy samochód dokładnie wmeldowany w mur. Dookoła oczywiście tłum gapiów. Szybkie przesłuchanie i wzywasz pomoc. Referujesz sprawę i kontyuujesz swoj patrol.

Od tej chwili rozpoczną się wydarzenia, które wstrząsną cichym Lytton. Nawet nie spodziewasz się, że to Ty, weteran policji, będziesz ich bohaterem.

Sonny Bonds uwikła się tak, że trudno będzie mu się pozbierać. Postarajmy się pomoc mu, za pośrednictwem komputera IBM i gry "POLICE QUEST".

Największa przyjemnosć gracza leży w samodzielnym rozwikłaniu gry. Dlatego też przerwę teraz opowiesc, podając tylko kluczowe informacje, bez których raczej sobie nie poradzicie. Zakładam, że znacie już "KING'S QUEST" — jeśli nie, nauczcie się dobrze obsługiwać "POLICE QUEST".

A oto skrót przygód policjanta, w postaci informacji kluczowych:

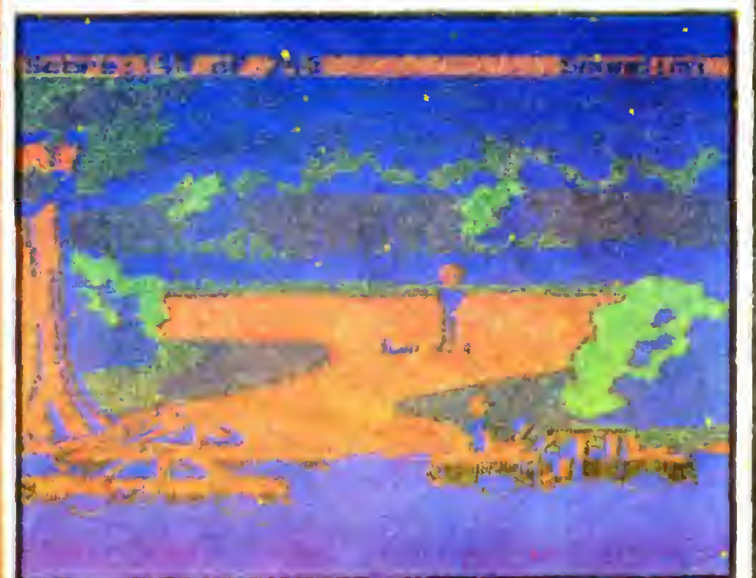
— Wciśnij Fl, by otrzymać sciągawkę klawiszy,
— Aby zatrzymac ścigany samochód, jedż za nim na sygnale,
— Piratkę drogową spisz i daj mandat, nie daj się uwieść, a jeśli już, to nie dzwoń potem do niej!
— Hippisow wykurzysz, każąc im zabrać motory. Gdy rzucą się do bitki, użyj pałki (jest w samochodzie na drzwiach),
— Pijaka zaaresztujesz dając mu probierz (TEST). Zakuj'go z tyłu!
— Aresztowanych wież do więzienia, zostaw broń w skrytce,
— Morderca z Cadillaca jest grozny, zawołaj pomoc, każ mu podnieść ręce, położyć się i zakuj w kajdanki,
— Podanie do dywizji narkotycznej napisz na blankiecie w korytarzu,
— Gdy zatrudnią Cię już w Narkotycznej, przebierz się i korzystaj z biurka,
— W parku odbędzie się wymiana narkotyków; schowaj się za krzakiem, a gdy handlarze zaczna się kłocic, wyjmij bron, szepnij przez krótkofalówkę, że zaczynasz, krzyknij „HALT" i wybiegnij zza krzaka. Złapanego handlarza zakuj w kajdanki i wróć do samochodu. Nie rozmawiaj z aresztantami!
— W więzieniu przekonaj Marię, by pomogła wam w operacji, wroć na posterunek,
— Ufarbuj włosy, przebierz się i jedź Cadillacem do hotelu Delphoria. Tam odegraj zaplanowaną wcześniej scenkę (o wszystkim zostaniesz poinformowany), wynajmij pokój i zadzwon z niego po taksówkę. Potem przez barmana dostan się do kasyna i wygraj partię pokera. Wróć do pokoju, wez od jednego z trzech kolegów radio ukryte w długopisie, i wróć do kasyna. Wygraj drugą partię i pójdz za Aniołem Smierci do pokoju. Przed pokojem użyj radia i śmiało wchodź. Dalej wypadki potoczą się same... ale to oczywiście niespodzianka.

Leniwemu graczowi ten skrót nie wystarczy do skonczenia "POLICE QUEST". Ale leniwi niech grają w "ASTEROIDS", to nie będzie dla nich zbyt trudne!

Dziękujemy Przemkowi, Tomkowi i Wojtkowi za pomoc w ukończeniu "POLICE QUEST".

Komputer: Na razie tylko IBM PC

Marcin i Mateusz Przasnyscy



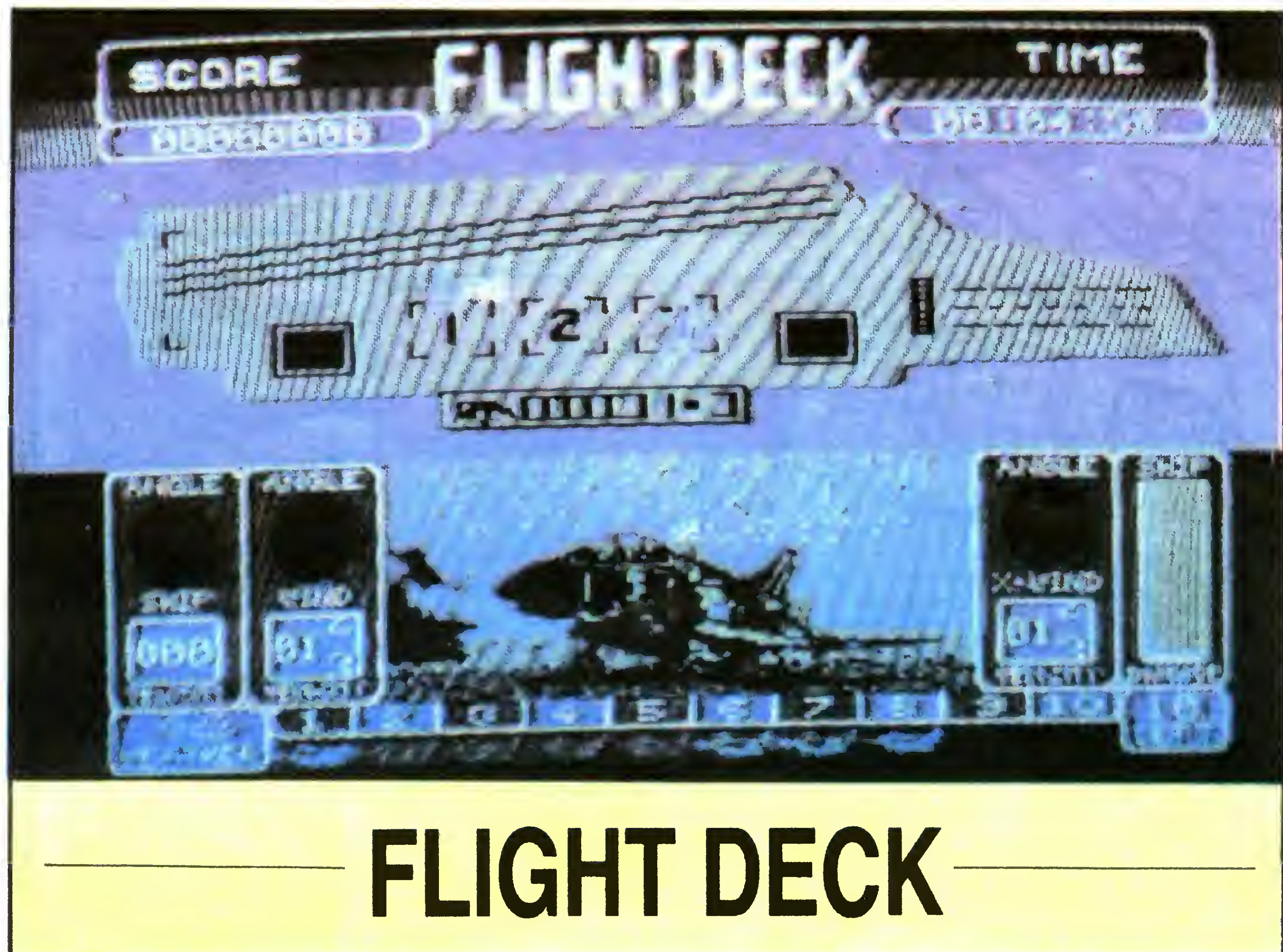
10 BAJTKOWA LISTA PRZEBOJÓW

Gryzor, który przez dwa miesiące trzymał się na pierwszym miejscu, spadł wreszcie prawie do końca. Natomias Platoon wciąż oscyluje w czołówce. W pozostałej części Listy widać przegrupowania, ale w istocie zmienia się niewiele. Virus, który pojawił się jak kometa, odleciał już do trzeciej dziesiątki i chyba szybko nie powróci. Na majowe notowanie Listy przebojów otrzymaliśmy 2780 propozycji pierwszej dzisiątki, głosowano na 144 tytuły.

		ATARI	AMSTRAD	COMMODORE	SPECTRUM
1	STREET FIGHTER	×	×	×	×
2	PLATOON	×	×	×	×
3	UNIVERSAL HERO	×	×	×	×
4	GREEN BERET	×	×	×	×
5	HOT SHOT		×	×	×
6	MOUSE TRAP	×	×	×	×
7	ARMY MOVES			×	×
8	KING'S QUEST	×	×	×	×
9	GRYZOR	×	×	×	×
10	STRONG MAN			×	×



Kopertę nadesłał Maciej Kubasik z Gliwic



FLIGHT DECK

John O'Brien był komandosem starej generacji. Nie uznawał cofania się przed wrogiem ani jedzenia czekolady. Guma w jamie ustnej była zawsze jego nieodłącznym towarzyszem. Pił tylko whisky i mocną czarną kawę.

Dorobił się majątku na giełdzie (nie bajtkowej) i postanowił zabawić się — niszcząc Nowy Jork. Wykupił niedużą wyspę na Atlantyku, zainstalował radary, zbudował lotnisko i olbrzymie hangary. Pod ziemią umieścił laboratoria, w których zaczął wytwarzać rakiety jądrowe.

Rząd amerykański, zdenerwowany doniesieniami o działalności Johna obudził Ciebie, mój sympatyczny Czytelniku i rozkazał Ci zniszczyć jego bazę. Do pomocy masz joystick oraz... supernowoczesny, największy, najszybszy, najdoskonalszy lotniskowiec marynarki USA.

Jest to oczywiście okręt typu NIMITZ, który posiada 2 reaktory atomowe, pozwalające osiągnąć prędkość 44 węzłów. Jest bardzo zwrotny i wyjątkowo długi (322 metry). Ma on niestety bardzo poważną wadę — nie posiada broni przeciwlotniczej, a więc jest zupełnie bezbronny dla samolotów przeciwnika.

Twój statek może zabrać na swój pokład tylko (!) 10 samolotów:

— 4 samoloty myśliwskie McDONNEL DOUGLAS F/A-18A HORNET. Są bardzo lekkie (12 ton), zwrotne i szybkie (1.8 Ma). Używaj ich do obrony lotniskowca.

— 4 samoloty myśliwsko-szturmowe McDONNEL DOUGLAS F-4E PHANTOM II. Jak na swoją dość dużą masę (14 ton), osiągają niesamowitą prędkość (2.4 Ma). Bombarduj nimi wyspę.

— 2 samoloty szturmowe VOUGHT A-7E CORSAIR II. Z pełnym uzbrojeniem i zapasem paliwa są bardzo ciężkie (19 ton) i osiągają dosyć małe prędkości (ok. 0.9 Ma). Niezbędne są do przeprowadzenia decydującego ataku.

„Lotniskowiec zrobił zwrot w lewo i poszedł pod wiatr północny. Zapalone lampki startowe drgały od wibracji maszyn — kończono grzać silniki samolotów. Zaryczał silnik czołowej maszyny myśliwskiej. Wśród kołysań kadłuba samolot rozpoczynał swój bieg, zrazu powoli, potem z coraz większą prędkością. Pierwszy samolot uniósł się pomyślnie w powietrze. Następne maszyny podążały za nim."

W taki sposób startowały samoloty podczas wojny na Pacyfiku. Zmieniły się jednak czasy, a także technika startu i lądowania. Przede wszystkim musisz obrócić swój lotniskowiec pod wiatr. Manewr ten jest praktycznie gwarancją pomyślnego startu. Następnie wyprowadź jedną z maszyn i ustaw ją na pasie startowym (z przodu statku). Teraz wystarczy już utrzymać małą kropkę w centrum kwadratu i samolot z rykiem silników wzbija się w powietrze.

Dużo trudniejsze jest lądowanie. Musisz rozpędzić lotniskowiec do maksymalnej szybkości i ustawić go pod wiatr. Teraz naprowadź maszynę nad statek i oczekuj sygnału pozwalającego na lądowanie. Rola pilota sprowadza się do utrzymania właściwego kursu. Pomaga w tym system ILS widoczny u dołu ekranu. Dokładne podejście do pasa kończy się wyhamowaniem samolotu na kauczukowych gumach.

Gra rozpoczyna się 15 mil od wyspy. Radary przeciwnika znajdujące się na niej mają zasięg około 9—10 mil. Przekroczenie tej granicy spowoduje z całą pewnością zniszczenie lub poważne uszkodzenie twojego statku. Gra zakończy się także po straceniu wszystkich samolotów lub bombowców. Uważaj na paliwo, gdyż Twoje samoloty nie latają na powietrze.

Oto klawisze sterujące:
CRSR lewo-prawo — obracanie lotniskowcem
CRSR góra-dół — przyspieszanie i zwalnianie
F1 — lotniskowiec
F3 — mapa akwenu
F5 — mapa wyspy

Zakończenie tej gry jest także bardzo przyjemne. Cała wyspa zamienia się w parę wodną zmieszaną z glebą, a Ty odbierasz komunikat, że masz płynąć na sąsiednią wyspę, gdzie zdążył przemieścić się O'Brien. Życzę przyjemnej zabawy.

Komputer: Commodore 64/128

Luke

KRÓL I KRÓLOWA — GIER —



Magda Paradowska
klasa VI D SP nr 57
Komputer: Spectrum
Ulubione gry: „Bombonn", „West Bank"
Hobby: kwiaty, zwierzęta (papuga, pies, myszka chińska)



Mirosław Mamczur lat 6
Komputer: Atari
Ulubione gry: „Green Beret", „Hacker"
Hobby: lubi mamy czytanie i rysowanie



SUPER HUEY

Super Huey to kod, pod którym kryje się wspaniały śmigłowiec typu Sikorsky S-70 Black Hawk. Jest to jeden z najnowocześniejszych helikopterów wielozadaniowych produkowanych w USA. Potężny, czteropłatowy, tytanowy wirnik o średnicy 16 metrów pozwala mu osiągnąć prędkość 300 km/h. Dzięki tej zalecie jest praktycznie niezniszczalny — atakuje i znika niezauważony. Uzbrojony w 2 karabiny maszynowe oraz cztery wyrzutnie pocisków rakietowych, może atakować właściwie wszystko, od słomianych chat do przeciwpancernych bunkrów.

Kabina pilota obłożona jest dźwiękoszczelną masą pokrytą superelastycznym materiałem. Główny komputer obsługuje wszystkie urządzenia wewnętrzne, a także działka i rakiety. Obroty wirnika są automatycznie regulowane, start i lądowanie wykonuje praktycznie komputer — rola pilota ogranicza się do wciskania klawiszy i ruszania drążkiem sterowym. Podwozie jest automatycznie chowane, koła są odporne na pociski (z wyjątkiem granatników). W sumie Super Huey jest niezniszczalny.

To właśnie taką maszyną będziesz sterował, gdy wgrasz na swój komputer tę grę. Masz do wyboru cztery misje: SOLO FLIGHT (samotny lot), COMBAT (walka), RESCUE (ratunek), EXPLORE (wywiad). Każda z nich pozwala na wspaniałą zabawę a także naukę. Musisz pamiętać o jednym — nie wolno się cofać. Ucieczka najczęściej kończy się fatalnie.

Sterowanie helikopterem:
JOYSTICK PORT II
LEWO + FIRE — otwarcie przepustnicy
PRAWO + FIRE — zamknięcie przepustnicy
F1 — włączenie komputera pokładowego
F3 — włączenie silnika
F5 — włączenie wirnika
F7 — wyłączenie wszystkich urządzeń

Inne funkcje wywołuje się za pomocą kodów wpisywanych z klawiatury. Podaję je wraz z funkcją:
ABT — koniec misji
ACS — automatyczne ustalenie kursu
CLM — warunki pogodowe
AT — dystans do celu
GTK — mapa
HOM — odległość od lotniska
LAR — ładowanie rakiet i amunicji (tylko na lotnisku)
MAC — uaktywnienie działek i wyrzutni rakiet
POW — włączenie wszystkich urządzeń pokładowych
RAD — włączenie radaru
TRK — kurs radarowy
UOR — nawigacja
USI — szybkość helikoptera
XXX — odwołanie ostatniego rozkazu

Dla początkujących opiszę, jak powinien wyglądać start i lądowanie.

START:
— włącz komputer
— włącz urządzenia pokładowe
— uruchom silnik
— używając przepustnicy ustaw moc silnika na 1700
— włącz wirnik, poczekaj aż osiągnie 1700 obr./min.
— zwiększ moc silnika do 3000
— stery pionowe (JOYSTICK lewo) podnieś „do góry"

LĄDOWANIE:
— podleć jak najbliżej do lądowiska
— zredukuj szybkość do zera
— obniż się na 30—40 metrów
— ustaw się dokładnie nad celem
— zmniejsz moc silnika do 2500
— obniż się na 2—3 metry
— wyłącz wszystkie urządzenia (F7)

Gra nie jest wybitna pod względem muzycznym i graficznym. Ma wiele wad np. nie da się rozbić helikoptera, niektóre opcje zawieszają się. Lecz obok GUNSHIP-a jest to nadal jedna z najlepszych symulacji śmigłowca. Polecam ją wszystkim początkującym w grach symulacyjnych, a także prawdziwym zawodowcom, którzy startują po jednym pociągnięciu drążka.

Komputer: ZX Spectrum 48/+, Commodore 64, Atari XL/XE

Luke

CHIPWAR — SUPLEMENT

Czy macie już dosyć gry „CHIPWAR"? Czy dostajecie drgawek na dźwięk jej tytułu? A może w ogóle skasowaliście ją ze swoich zbiorów. Teraz macie szansę powrócić do niej i wreszcie naprawić swój komputer.

Wewnątrz układu scalonego, podczas jego naprawiania widoczne są w prawym dolnym rogu trzy litery. Środkowa z nich, „S" służy do podejrzenia prawidłowego schematu naprawianej części. Każdorazowe podejrzenie odbiera jednak 100 jednostek czasu. Postąpcie więc następująco: wciskajcie „S" (oczywiście na ekranie) tak długo, aż licznik czasu wskaże liczbę większą od stu o kilkadziesiąt. Wtedy wciśnijcie „S" ponownie i nie zwalniajcie przycisku FIRE. Gdy czas skończy się, układ będzie naprawiony, a to na skutek drobnego błędu w programie gry „CHIPWAR".

W ten sposób naprawić można wszystkie elementy po kolei i rozkoszować się widokiem napisu „GAME OVER" kończącego tę grę, tym razem definitywnie.

Gen

S.O.S.

Jestem użytkownikiem Atari 65XE. Szukam opisów do gier: CHIMERA, THE GOONIES, NINJA. Jak przejść strzały w grze WORLD KARATE? Jak wpisywać nieśmiertelności na ten komputer?
Kornel Rydzewski ul. Noniewicza 10 b m 49 16-400 Suwałki

Szukam dokładnego opisu (z miejscem, gdzie znajduje się rakieta) do gry MERCENARY oraz informacji, czy programy Turbo Esprit, Ghost Goblins istnieją na Atari ze stacją dysków.
Krzysztof Karkosza Wzg. Krzesławickie 14 a m 37 31-720 Kraków

Nie mogę uruchomić gry SUBMARINE COMANDER. O co chodzi w grze BLUE THUNDER? Komputer Atari 65 XE.
Adam Ortowski ul. Krajewskiego 12 a m 1004 45-256 Opole

Jak uzyskać i wpisać do komputera Amstrad CPC 464 nieśmiertelność do gier: RAMBO, FIRE LORD, COMMANDO, GHOST'N GOBLINS, JET SET WILLY?
Jacek Białkowski ul. Mała 9 05-410 Józefów 3

Mam Spectrum 48. Proszę o opisy programów: ELITE, ZORRO, BATMAN, SILENT SERVICE, WAR OF THE WORLDS, ponieważ mam z nimi kłopoty.
Konrad Wojczyszyn ul. Berlinga 3 a m 15 66-400 Gorzów Wlkp.

Proszę o pomoc w następujących grach: TIRES — nie wiem, jak przejść trzecie jeziorko, EUROPE THEATRE — nie znam szyfru na granie chemiczne i nuklearne, PRELIMINARY MONTY — jak skończyć tę grę. Wszystkie na Atari 800 XL.
Piotr Kozimor ul. Gdańska 28 m 2 83-047 Przywidz woj. gdańskie

O co chodzi w grach: JACK THE NIPPER II, JETMAN! SPY HUNTER? Chcę wymienić oprogramowanie na ZX Spectrum.
Robert Grochal ul. Korczaka 5 m 29 Bydgoszcz

Proszę o dokładny opis gry SPY VS SPY I na Atari 65 XS.
Bartłomiej Sobaś ul. Mała Góra 16 m 141 30-864 Kraków

Pilnie szukam następujących gier na Atari POP EYE, BARBARIAN, KUNG FU MASTER, WORLD KARATE CHAMPIONSHIPS. W zamian inne.
Paweł Teplik Os. Rzeczypospolitej 14 m 122 61-391 Poznań

W zamian za opis do gry UNIVERSAL HERO odstąpię kilka nowych gier. Szukam też gier: BARBARIAN i GAUNTLET. Komputer Atari 800 XL.
Jarosław Grzelak ul. Wierzbowa 2 G m 11

Jak uzyskiwać nieśmiertelność w grach na Atari. Proszę też o opis gry BLUE MAX.
Robert Mrowiński ul. Belwederska 4 m 1 60-275 Poznań

Kto mi pomoże w grach: BLUE MAX, ROAD RACE, SPY Vs SPY II, BEACH HEAD II na Atari 800 XL?
Tomasz Hasiuk Miluki 19-325 Straduny woj. suwalskie

Jak zdobyć program do gry PHOENIX? Komputer Spectrum.
Grzegorz Zaczkowski ul. Lotników 1 m 46 78-520 Złocieniec

Przyślijcie mi opisy gier: MOLECULE MAN, SILENT SERVICE, MASTER OF THE LAMPS, FRONT w wersji kasetowej na Atari 65 XE.
Mariusz Besaraba ul. Kmicica 9 43-382 Bielsko-Biała

Mam Atari 65 XE. Proszę o opisy gier BROADSIDES i B-L NUCLEAR BOMBER.
Marcin Loba ul. Górnicza 5 m 11 62-510 Konin

Potrzebuję pomocy w grach ZORRO i MONTE 16 k. Kto może mi przysłać plan i opis do nich? Interesuje mnie także gra PANAMA JOE. Komputer Atari 130 XE.
Robert Szczepkowski ul. Ilińska 3 14-140 Miłomłyn

W zamian za gry symulacyjne i taktyczne, a szczególnie SOLO FLIGHT — opisy do innych. Komputer Atari 800 XL.
Bogdan Stech ul. Spokojna 7 76-200 Słupsk

Jak zniszczyć toster w HIMERZE i proszę o nieśmiertelność do GIRUS-a. Mam Atari.
Wojciech Zator Zabierzów ul. Myszala 14 woj. krakowskie

Sukam gier: JET SET WILLY II, DYNAMITE DAN II i JACK THE NIPPER. Spectrum+.
Ania Świderska ul. Smutna 26 m 33 91-729 Łódź

Jak zebrać: but, podkowę i do czego służy klucz w grze ZORRO? Jak pokonać ósmego przeciwnika w grze BARBARIAN? Wszystkie na C-64.
Robert Grosiak Wężyki 96-514 Rybno woj. skierniewickie

Przyślijcie, proszę, opis do gry C'EST LA VIE na Atari 65 XE.
Marcin Główka ul. 18-go Listopada 18 m 9 41-250 Czeladź woj. katowickie

Usilnie szukam gry BARBARIAN z opisem a także sposobów uzyskiwania nieśmiertelności za pomocą programu NEW COPY. Może ktoś ma opis do gry RED MAX na Atari 65 XE? W zamian programy i opisy.
Maciej Smektała ul. Okrężna 3 64-600 Oborniki

KOPIOWANIE Z ATRYBUTAMI

czyli przygód z drukarką część druga

Czy pamiętacie moje ostatnie kłopoty? Wyobraźcie sobie, że nie skończyły się one wcale w momencie napisania programu kopiującego ekran w trybie graficznym. Przypominam — wszystko zaczęło się od pożyczenia drukarki Seikosha GP-500AS. Jak wiadomo, apetyt rośnie w miarę jedzenia. Zapragnąłem skopiować ekran tak, by było widać, że jest kolorowy.

Znacie Art Studio? Jest to taki program do rysowania. Teoretycznie pośród jego wielu możliwości figuruje opcja PRINT, umożliwiająca wydrukowanie zawartości ekranu w taki sposób, by różnym kolorom punktów na ekranie odpowiadały różne kombinacje punktów na papierze (jeden punkt ekranu odpowiada wtedy na papierze kwadratowi składającemu się — w zależności od możliwości drukarki — z czterech lub dziewięciu punktów). Napisałem „teoretycznie", bo jak wiadomo komputery wcale nie ułatwiają życia swoim właścicielom. Zawsze znajdzie się powód, dla którego nie da się zrobić tego, co **teoretycznie** powinno dać się zrobić bez żadnego problemu. Tak samo było tym razem. Pogięła się taśma magnetofonowa. Samo Art Studio dało się wczytać, ale już program instalujący je dla różnych konfiguracji sprzętowych — nie.

Ci z Was, którzy czytali ostatni artykuł, już wiedzą czym to się skończyło i następnego zdania mogą nie czytać. Jak to zwykle w takich sytuacjach bywa, musiałem sam napisać program robiący to, co było mi potrzebne. Niestety, tym razem problem nie da się rozwiązać w taki sam sposób jak poprzednio. Żeby skopiować ekran punkt po punkcie, wystarczyło go pokroić na paski o szerokości siedmiu pikseli i każdy taki pasek wysłać na drukarkę. Teraz ekran musiałby być pokrojony na paski o szerokości trzech i pół pikseli (możliwości Seikoshy pozwalają tylko na zastąpienie każdego punktu kwadratem dwa na dwa), i cała operacja byłaby bardzo skomplikowana. Zabrałem się więc do pracy w inny sposób. Pokroiłem na paski to, co miało być na wydruku — łatwo jest sprawdzić, czy punktowi na papierze odpowiada na ekranie punkt o kolorze tła, czy o kolorze tuszu.

Początkowo miałem zamiar napisać program w Basic-u, żeby sprawdzić algorytm, ale bardzo szybko zrezygnowałem. Pierwsze próby wykazały, że na kompletny wydruk trzeba by czekać kilka godzin. Dlatego też zdecydowałem się na sam program w kodzie maszynowym. Tak samo jak w poprzednim artykule, oprócz wydruku z asemblera, zamieszczam program ładujący w Basic-u — program nie jest relokowany, uruchamia się go przez **PRINT#4;USR 60000,** po otwarciu strumienia instrukcjami **OPEN#4; "B": FORMAT "B";9600.**

Marcin Borkowski

```
 10 CLEAR 59999: LET r=60000
 20 FOR i=r TO r+200 STEP 10
 30 LET s=0
 40 FOR j=0 TO 9
 50 READ a: POKE i+j,a
 60 LET s=s+a
 70 NEXT j
 80 READ a: IF a<>s THEN PRINT "Błąd w lini ";100+i-r
 90 NEXT i
100 DATA 221,33,38,235,33,0,0,34,39,235,868
110 DATA 62,8,215,33,0,0,34,43,235,221,851
120 DATA 54,0,0,42,39,235,6,0,221,78,675
130 DATA 0,167,237,74,34,41,235,203,60,203,1254
140 DATA 29,77,42,43,235,203,60,203,29,69,990
150 DATA 205,170,34,71,4,126,7,16,253,230,1116
160 DATA 1,79,124,15,15,15,230,3,246,88,816
170 DATA 103,126,203,41,56,3,15,15,15,230,807
180 DATA 7,71,33,2,235,213,17,4,0,4,586
190 DATA 237,90,16,252,209,58,41,235,230,1,1369
200 DATA 40,2,35,35,58,43,235,230,1,40,719
210 DATA 1,35,126,31,203,26,221,52,0,221,916
220 DATA 126,0,254,7,32,153,122,55,31,215,995
230 DATA 42,43,235,35,62,95,189,32,133,124,990
240 DATA 31,48,129,62,10,215,42,39,235,6,817
250 DATA 7,35,16,253,62,255,189,32,3,62,914
260 DATA 1,188,194,103,234,201,1,1,1,1,925
270 DATA 1,0,1,1,1,1,0,1,0,1,7
280 DATA 1,0,1,0,0,1,0,0,1,0,4
290 DATA 0,1,0,0,0,0,0,0,0,0,1
300 DATA 0,0,0,0,0,0,0,0,0,0,0
```

```
10          ORG     40000
20          LD      IX,IVAR
30          LD      HL,0
40   MLP3   LD      (X),HL
50          LD      A,8
60          RST     #10
70          LD      HL,0
80   MLP2   LD      (YPAP),HL
90          LD      (IX+0),0
100  MLP1   LD      HL,(X)
110         LD      B,0
120         LD      C,(IX+0)
130         AND     A
140         ADC     HL,BC
150         LD      (XPAP),HL
160         SRL     H
170         RR      L
180         LD      C,L
190         LD      HL,(YPAP)
200         SRL     H
210         RR      L
220         LD      B,L
230         CALL    #22AA
240         LD      B,A
250         INC     B
260         LD      A,(HL)
270  LPP1   RLCA
280         DJNZ    LPP1
290         AND     1
300         LD      C,A
310         LD      A,H
320         RRCA
330         RRCA
340         RRCA
350         AND     3
360         OR      #58
370         LD      H,A
380         LD      A,(HL)
390         SRA     C
400         JR      C,INK
410         RRCA
420         RRCA
430         RRCA
440  INK    AND     7
450         LD      B,A
460         LD      HL,DANE-4
470         PUSH    DE
480         LD      DE,4
490         INC     B
500  LPP2   ADC     HL,DE
510         DJNZ    LPP2
520         POP     DE
530         LD      A,(XPAP)
540         AND     1
550         JR      Z,LPP3
560         INC     HL
570         INC     HL
580  LPP3   LD      A,(YPAP)
590         AND     1
600         JR      Z,LPP4
610         INC     HL
620  LPP4   LD      A,(HL)
630         RRA
640         RR      D
650         INC     (IX+0)
660         LD      A,(IX+0)
670         CP      7
680         JR      NZ,MLP1
690         LD      A,D
700         SCF
710         RRA
720         RST     #10
730         LD      HL,(YPAP)
740         INC     HL
750         LD      A,95
760         CP      L
770         JR      NZ,MLP2
780         LD      A,H
790         RRA
800         JR      NC,MLP2
810         LD      A,10
820         RST     #10
830         LD      HL,(X)
840         LD      B,7
850  MLP4   INC     HL
860         DJNZ    MLP4
870         LD      A,255
880         CP      L
890         JR      NZ,MLP5
900         LD      A,1
910         CP      H
920  MLP5   JP      NZ,MLP3
930         RET
940  DANE   DEFB    1,1,1,1
950         DEFB    1,0,1,1
960         DEFB    1,1,0,1
970         DEFB    0,1,1,0
980         DEFB    1,0,0,1
990         DEFB    0,0,1,0
1000        DEFB    0,1,0,0
1010        DEFB    0,0,0,0
1020 IVAR   DEFB    0
1030 X      DEFW    0
1040 XPAP   DEFW    0
1050 YPAP   DEFW    0
```

STATYSTYKA KALKULATORA

Obecnie większość kalkulatorów naukowych czy programowych wyposażona jest przez producenta w statystykę dwóch zmiennych, a przynajmniej jednej zmiennej.

Ze statystyką związane są w kalkulatorach klawisze oznaczane tajemniczą sigmą Σ z przeróżnymi dodatkami. Celem niniejszego artykułu nie jest nauczenie Was podstaw statystyki, ale wyjaśnienie sposobów ułatwiania sobie życia przy korzystaniu z gotowych funkcji statystycznych. Zaskoczony bowiem byłem, gdy zobaczyłem, ile to osób znających podstawy statystyki nie potrafi policzyć podstawowych parametrów rozkładów statystycznych, takich jak odchylenie standardowe czy współczynnik korelacji. Sprawa leży często w nieznajomości odpowiednich wzorów, ale częściej w nieumiejętności posługiwania się paroma klawiszami.

W większości kalkulatorów ze statystyką związanych jest kilka rejestrów (około 6). Wprowadzenie danych statystycznych do kalkulatora zależne jest od modelu, ale na ogół polega na wciskaniu po każdej liczbie klawisza oznaczonego Σ+. Po wciśnięciu tego klawisza kalkulator przeprowadza automatycznie szereg operacji: zwiększa o jeden zawartość licznika wprowadzonych danych (klawisz oznaczony przez n lub Σn); ponadto zakładając, że liczby wprowadzane są parami w formacie x,y, to x dodawany jest do rejestru przechowującego sumę kolejnych x-ów (oznaczany przez Σx), kwadrat x dodawany jest do rejestru przechowującego sumę kwadratów x-ów (Σx^2). Te same operacje dokonywane są na wartościach y. Na koniec iloczyn x i y dodawany jest do rejestru przechowującego sumę iloczynów tych dwóch zmiennych (Σxy). Dla kalkulatorów ze statystyką jednej zmiennej działania są ograniczone przeważnie do operacji Σx, Σx^2,n.

Ile danych można w ten sposób wprowadzić? Na ogół nieograniczoną ilość, gdyż jak widać wszystkie podstawowe operacje statystyczne dokonywane są na bieżąco na wprowadzanych liczbach. Rzadko ograniczeniem jest przepełnienie rejestru nie mogącego przechowywać liczby z zakresu np. 0—99. Po wprowadzeniu danych możemy żądać od kalkulatora wyświetlenia zawartości odpowiednich wyżej wymienionych rejestrów. Możemy też wyświetlić parametry rozkładu statystycznego, które to kalkulator oblicza i wyświetla błyskawicznie w oparciu o zawartość rejestrów statystycznych i odpowiednią formułę matematyczną np. odchylenie standardowe oblicza on jako $s=\sqrt{((\Sigma x^2 - nx^2)/(n-1))}$, gdzie x=Σx/n i oznacza średnią arytmetyczną wartości wprowadzonych x-ów. Często na klawiaturze występują klawisze tak właśnie oznaczane. Podobnie się dzieje z pozostałymi różnorodnymi parametrami rozkładu statystycznego obliczanymi w oparciu o dane z rejestrów.

Oprócz kalkulatorów z wyżej wymienioną statystyką istnieją kalkulatory ze statystyką kumulacyjną (są to niektóre kalkulatory programowalne i programowalne). Działanie takich kalkulatorów polega na tym, że do pamięci wprowadzana jest macierz z danymi. Ilość kolumn tej macierzy charakteryzuje ilość zmiennych (często więcej niż 2, a nawet do 15). Na ogół jedynym ograniczeniem w takich kalkulatorach, jeśli chodzi o ilość danych i liczbę zmiennych, jest dostępna pamięć. Dopiero po wprowadzeniu wszystkich danych i wciśnięciu klawisza np. Σx następuje obróbka danych, w tym wypadku sumowanie wartości w poszczególnych kolumnach macierzy. Pewną wadą tej metody jest fakt, że dane są nieobrobione wstępnie i dopiero po wciśnięciu klawisza funkcyjnego przeprowadzane są obliczenia, co przy sporej ilości danych może zająć trochę czasu. Wady tej nie possiadają kalkulatory z pierwszej omówionej grupy. Zaletą kalkulatorów ze statystyką kumulacyjną jest całkowita dowolność w programowaniu obliczeń różnych funkcji statystycznych w oparciu o przechowywaną w pamięci macierz i dostępną w całości w każdej chwili.

Oprócz wspomnianych już klawiszy występują w kalkulatorach klawisze służące do poprawiania, kasowania danych oznaczane jako CD lub Σ- oraz zerowania rejestrów.

Najczęściej w szkołach i na wyższych uczelniach statystykę kalkulatorową wykorzystuje się przy dobrze krzywej do danych doświadczalnych. Chodzi tu o dopasowanie wybranych funkcji (linia prosta, funkcje: potęgowa, wykładnicza i logarytmiczna) do zebranych danych doświadczalnych dla jednej zmiennej niezależnej x i zmiennej zależnej y. Typowym testem określającym prawidłowość tego dopasowania jest współczynnik korelacji r (tak właśnie oznaczany na klawiaturach kalkulatorowych). Wartości jego są ilościową miarą związku (stopnia powiązania) pomiędzy dwoma zmiennymi. Jeżeli wartości r są bliskie +/−1, to oznacza to prawie idealne powiązanie zmiennych, r ≅ 0 oznacza prawie całkowicie przypadkowy związek zmiennych.

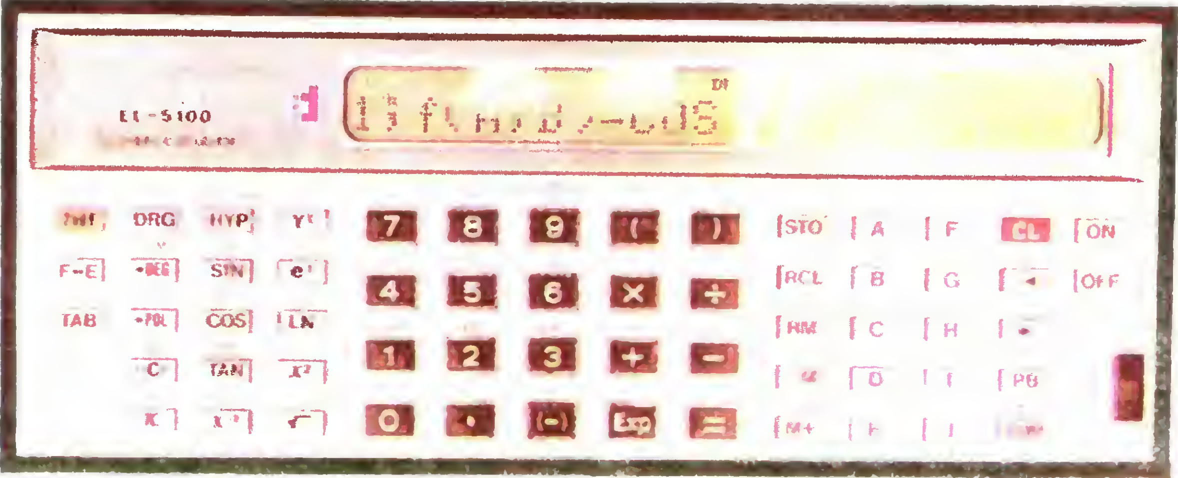
Kalkulatory standardowo przystosowane są na ogół do dopasowywania danych do linii prostej y=ax+b (w niektórych kalkulatorach np. SHARP często się zdarza, że linia ta ma równanie y=bx+a). W związku z tym na klawiaturze pojawiają się dwa kolejne klawisze oznaczane przez a i b. Po wprowadzeniu danych i wciśnięciu tych klawiszy otrzymujemy w rezultacie odpowiednio tangens kąta pochylenia prostej do osi ox i wartość rzędnej przecięcia osi oy przez prostą. Współczynnik korelacji wskazuje na prawidłowość wyboru postaci funkcji. Jeżeli r znacznie odbiega od +/−1 (często dane układają się zgodnie z krzywymi wykładniczymi lub logarytmicznymi) należy dane dopasować do innych funkcji. Jak to jednak zrobić, gdy funkcje te nie są wbudowane do kalkulatora? W tym celu można posłużyć się podanymi w tabelce wzorami. Pierwsza kolumna podaje postać funkcji, do jakiej będziemy dopasowywać dane. Następnie podane są wzory współczynnika korelacji. Wszystkie obliczenia można, mimo skomplikowanej postaci wzorów, przeprowadzić błyskawicznie w oparciu o parę klawiszy typu Σx, Σxy itp. Oznaczenie Σlny oznacza, że do kalkulatora wprowadzamy wartość y, obliczamy logarytm naturalny tej wartości i dopiero wprowadzamy tak obliczoną wartość do rejestrów statystycznych przez wciśnięcie klawisza Σ+. Obliczenia wartości współczynników należy rozpoczynać od znalezienia b1.

Warto zauważyć, że w przypadku dysponowania kalkulatorem programowalnym lub komputerem można napisać program na obliczanie kolejno wartości współczynników korelacji r dla wszystkich zamieszczonych postaci funkcji, a dopiero na ich podstawie wybrać postać funkcji, która najlepiej pasuje do zestawu danych doświadczalnych.

Przedstawione elementy statystycznej obróbki danych eksperymentalnych w żadnym stopniu nie wyczerpują problemów, które można efektywnie rozwiązać posługując się minikalkulatorem czy komputerem. Jest to raczej ilustracja możliwości bezpośredniego użycia zaprogramowanych w kalkulatorze instrukcji obliczeń statystycznych i zachęta do poznawania ukrytych możliwości kalkulatorów.

Dominik Falkowski

Postać funkcji	Współczynnik bo	Współczynnik b1	Współczynnik korelacji r^2
y = bo + b1x	$\bar{y} - b1\bar{x}$	$\frac{(\Sigma xy - \bar{x}\Sigma y)}{(\Sigma x^2 - \bar{x}\Sigma x)}$	$\frac{(\Sigma xy - \bar{x}\Sigma y)^2}{(\Sigma x^2 - \bar{x}\Sigma x)(\Sigma y^2 - \bar{y}\Sigma y)}$
$y = boe^{b1x}$ yi > 0	$\exp(\Sigma \ln y/n - b1\bar{x})$	$\frac{(\Sigma(x\ln y) - \bar{x}\Sigma \ln y)}{(\Sigma x^2 - \bar{x}\Sigma x}$	$\frac{(\Sigma(x\ln y) - \bar{x}\Sigma \ln y)^2}{(\Sigma x^2 - \bar{x}\Sigma x)(\Sigma(\ln y)^2 - (\Sigma \ln y)^2/n)}$
y = bo+b1lnx xi > 0	$(\Sigma y - b1\Sigma \ln x)/n$	$\frac{\Sigma(y\ln x - \bar{y}\Sigma \ln x}{\Sigma(\ln x)^2 - \Sigma(\ln x)^2/n)}$	$\frac{(\Sigma(y\ln x) - \bar{y}\Sigma \ln x)^2}{(\Sigma(\ln x)^2 - (\Sigma \ln x)^2/n)(\Sigma y^2 - \bar{y}\Sigma y)}$
$y = box^{b1}$ xi > 0 yi > 0	$\exp(\Sigma \ln y/n - b1(\Sigma \ln x)/n)$	$\frac{\Sigma(\ln x \cdot \ln y) - (\Sigma \ln x)(\Sigma \ln y)/n}{\Sigma(\ln x)^2 - (\Sigma \ln x)^2/n}$	$\frac{(\Sigma(\ln x \ln y) - (\Sigma \ln x)(\Sigma \ln y)/n)^2}{(\Sigma(\ln x)^2 - (\Sigma \ln x)^2/n)(\Sigma(\ln y)^2 - (\Sigma \ln y)^2/n)}$



MINIKALKULATOR SHARP EL-5100

Minikalkulator EL-5100 firmy SHARP jest ciekawym rozwiązaniem z uwagi na specyficzne walory funkcjonalne.

Jest to w założeniu bardzo wyspecjalizowany kalkulator programowalny przeznaczony do jak najsprawniejszego obliczania funkcji matematycznych zapisanych w postaci równania. Kalkulator posiada algebraiczny system nawiasowy rozwiązany moim zdaniem perfekcyjnie. Jak pisałem miesiąc temu o systemach operacyjnych, wspominałem o ogólnej filozofii nawiasowych systemów algebraicznych. Pisałem również, że większość z takich kalkulatorów posiada instrukcje wykonane tak jak w Odwrotnej Notacji Polskiej. Ten kalkulator jest chlubnym wyjątkiem. Wszystkie operatory wprowadzone są identycznie jak na kartce papieru. Można więc przepisywać problemy z książki nie zastanawiając się nad żadnymi dodatkowymi przekształceniami.

Kalkulator wyglądem przypomina tabliczkę czekolady o wymiarach 175x70x9,3 mm. Posiada duży 32-znakowy ekran ciekłokrystaliczny umożliwiający wpisanie 80 znaków w wierszu (scrolling). Ekran ten daje możliwość wpisania całego wyrażenia i dopiero wykonanie wszystkich działań. Minikalkulator nie posiada w ogóle instrukcji związanych ze złożoną programowalnością a w szczególności instrukcji skoków warunkowych, adresowania pośredniego itp. Wyposażony jest natomiast w bogaty komplet instrukcji obliczania funkcji elementarnych (również hiperbolicznych) i statystycznych. Kalkulator może pracować w trzech trybach. Pierwszy to tryb programowania, w którym wprowadza się formuły matematyczne. Można jednorazowo wprowadzić szereg takich formuł. Warunkiem jest, by łączna suma znaków nie przekraczała pojemności 80 rejestrów z maksymalną liczbą nawiasów nie większą niż 15. Drugi tryb to tryb kalkulatorowy, w którym dokonuje się wszystkich standardowych obliczeń. W tym trybie wywołuje się i oblicza w formie dialogowej również formuły zapisane w poprzednim trybie. Ostatni tryb pracy to pełna statystyka dwóch zmiennych, z regresją liniową i bogatym zestawem parametrów rozkładów statystycznych.

Wskaźnik ciekłokrystaliczny o mozaikowej budowie znaków umożliwia wyświetlenie dziesięciu liter (od A do J), cyfr oraz wszystkich słów kluczowych. Na korzyść tego wyświetlacza trzeba dodać, że jest on doskonale czytelny nawet przy słabym zewnętrznym oświetleniu, a jak wiadomo różnie to bywa z wyświetlaczami na ciekłych kryształach. Kalkulator zużywa minimalne ilości prądu. Osobiście używałem go intensywnie przez 6 lat i ani razu nie zmieniałem w nim baterii. Zasilany jest trzema bateriami LR44.

Z ciekawszych rozwiązań należy wspomnieć o funkcji PLAY BACK. W przypadku popełnienia jakiegoś błędu lub chęci powtórzenia obliczeń w pełnym ciągu wprowadzonym poprzednio (z ewentualnymi drobnymi zmianami) wciśnięcie klawisza oznaczonego przez PB spowoduje powrót do formuły matematycznej wprowadzonej poprzednio. Kolejnymi klawiszami INS i DEL (znanymi z komputerów) można następnie przeprowadzić edycję takich formuł co wydatnie przyczynia się do skrócenia toku obliczeń i uprzyjemnienia go.

Kalkulator wyświetla wyniki jako 10 znakową mantysę i 2 znakową cechę. Obliczenia przeprowadzane są wewnętrznie w oparciu o 12 cyfr mantysy i 2 cyfry cechy. Posiada 11 pamięci z czego jedna jest o dostępie swobodnym. W sumie kalkulator posiada około 98 funkcji wbudowanych a obejmujących poza już wspomnianymi funkcjami także kombinacje, permutacje, zmiany między różnymi układami współrzędnych i wiele innych.

Kalkulator mogę z czystym sumieniem polecić jako doskonały instrument wspomagający obliczenia wszystkim od 5 klasy szkoły podstawowej wzwyż. Jeżeli nie mamy zamiaru programować lub w tym celu stosujemy wyłącznie komputer to EL-5100 może zapewnić wiele lat przyjemnej pracy.

Dominik Falkowski

maxell®

ABC Data

MAXELL

Średnica	Zapis	Pojemność	DM
5 1/4"	2D	360 kB	1,75
5 1/4"	2HD	1.2 MB	3,75
3"	CF-2	360 kB	4,75

* Dyskietki 3" do mikrokomputerów Amstrad

Dyskietki magnetyczne Maxell odpowiadają najwyższym światowym standardom zapisu danych. Stosowanie nośników Maxell oznacza pewność i bezpieczeństwo danych.



MINI FLOPPY DISK
DOUBLE SIDE
DOUBLE DENSITY
SOFT SECTOR
5¼ INCH 48 TPI
MD2D

kasety do drukarek

* kasety do drukarek firmy EPSON

Typ drukarki	DM
FX-1000	11,40
LX-800	9,20



* kasety do drukarek firmy STAR

Typ drukarki	DM
LC-10 (NX-1000)	9,00
NX-15 (ND-15/NR-15)	11,60
ND-10 (NR-10)	10,60
NB-15 (NB 24-15)	12,60

* kasety do drukarek firmy NEC

Typ drukarki	Cena kasety DM
P-6	9,00
P-7	10,30

TO MNIE BIERZE!



Średnica	Zapis	Pojemność	DM
5 1/4"	2D	360 kB	1,75
5 1/4"	2HD	1.2 MB	3,75

Dyskietki magnetyczne firmy SKC są bardzo wysoko cenione w świecie mikroinformatycznym ze względu na swoje doskonałe właściwości magnetyczne.

SKC należy do przodujących producentów nośników magnetycznych. Doskonałe zaplecze techniczne firmy oraz intensywne prace badawcze, dają produkt najwyższej jakości.

UWAGA

Zastrzegamy sobie prawo zmiany cen. Ceny podane są bez kosztów transportu z Hamburga lub Berlina. Doliczana przez nas suma 40 DM pokrywa opłaty transportowe, odprawę celną oraz ubezpieczenie. W przypadku mniejszych zakupów (kasety, pisaki) opłata ta wynosi 20 DM. Za dostawę produktów o większych wymiarach (plotery A2, A0 itp.), kalkulujemy dopłatę wg cenników firmy-przewoźnika.

Zainteresowanych produktami innych firm prosimy o bezpośredni kontakt

C.ITOH houston PC arc AMPEX

Niniejszym zamawiam następujące artykuły:
. . . . sztuk. . . . DM. . . . . .
. . . . sztuk. . . . DM. . . . . .
. . . . sztuk. . . . DM. . . . . .
Transport (1 szt. DM 40) . . . . sztuk. . . . DM. . . . . .
Kwota pobierana przez bank . . . . DM 10,-
Razem: DM. . . . . .

W załączeniu kopia zlecenia bankowego na przelew w/w sumy na konto ABC Computersystems. Bank für Handel und Industrie BLZ: 100 800 00. Konto nr: 06 358 810 00. Konto dol.: 06 358 810 00 / 400.

W/w sprzęt proszę wysłać na adres:
Nazwisko i imię . . . .
Kod pocztowy i miejscowość . . . . tel. . . . . . .
Ulica i numer domu . . . .
Nazwisko, imię i adres zamawiającego (jeśli inne niż odbiorcy): . . . .
. . . .
data . . . . podpis . . . .

1. Po dokonaniu przelewu prosimy o wysłanie do nas załączonego zamówienia lub krótkiego listu (najlepiej na odwrocie kserokopii dowodu wpłaty) z dokładną informacją o tym, co Państwo zamawiacie i na jaki adres ma być wysłany.

2. Wyroby nasze mogą Państwo również kupić osobiście w Hamburgu lub zamówić wysyłkę z Bonn:

ABC Data GmbH
Ditmar-Koel-Str. 13
2000 Hamburg 11
RFN
tel. (040) 314003/3195874
tlx. 21 66 002 abc h
fax. (0049-40) 3191783
Dresdner Bank Hamburg
BLZ 200 800 00
Konto nr 061 612 61
Konto dol. 061 612 61

ABC Data GmbH
Postfach 200 146
Augustastr. 40
5300 Bonn 2, RFN
tel. (0228) 354480/90
tlx. 88 55 66 abcbs d
fax. (0049-228) 355635
Dresdner Bank Bonn 2
BLZ 370 800 40
Konto nr 2 688 475 00/400
Konto dol. 2 688 475 00/400

ABC Computersystems GmbH jest firmą zarejestrowaną w Amtsgericht, Berlin Charlottenburg, 92 HRB 26974
Dyrektor: Marian Pijarski



STRUKTURY DANYCH (1)

Osobom, których znajomość programowania wykracza poza kilka podstawowych instrukcji, pragnę zaproponować przeanalizowanie korzystania w programach z pamięci komputera.

Od strony technicznej możemy określić pamięć jako ciąg jednakowych komórek, z których każda może pomieścić pewną porcję informacji oraz ma jednoznaczny adres. Pomijamy w ten sposób wiele ważnych szczegółów technicznych, ale w naszych rozważaniach będą one potrzebne. Najprostszy schemat adresowania otrzymamy numerując wszystkie komórki kolejnymi liczbami całkowitymi, od zera poczynając. Często numery, czyli adresy komórek podajemy nie w systemie dziesiętnym lecz szesnastkowym.

Angielski skrót RAM zwykle tłumaczony (całkiem poprawnie) jako: „pamięć operacyjna" pochodzi od słów Random Access Memory i trochę bardziej dosłownie mógłby być tłumaczony jako: „pamięć o dostępie bezpośrednim". To tłumaczenie uwypukla istotną cechę pamięci operacyjnej współczesnych komputerów: dostęp do każdej z komórek zajmuje tyle samo czasu. Inaczej mówiąc, czas zapisania lub odczytania zawartości komórki nie zależy od jej położenia w pamięci. Czy w takim razie warto analizować odwołania programów do pamięci — przecież z technicznego punktu widzenia wszystkie są dokładnie takie same. Zgadza się, ale pisząc programy w języku wyższego rzędu, takim jak BASIC czy PASCAL zwykle nie korzystamy z fizycznych adresów komórek, lecz odwołujemy się do pamięci za pomocą konstrukcji tego języka. W kolejnych odcinkach postaram się pokazać na przykładach, że rozwiązania tego samego zadania będą się bardzo różniły w zależności od tego jak zostaną zorganizowane dane w pamięci operacyjnej — czyli od tego jakich konstrukcji użyjemy. Zarówno łatwość pisania programu jak też czas działania gotowego programu zależą nie tylko od wybranego algorytmu rozwiązania ale też od wybranych struktur danych.

Najprostszym z zapisywanych w pamięci obiektów są stałe i zmienne programu. Ich istota jest naturalna i oczywista: chcemy przechowywać wartości ustalonego typu, np. liczby całkowite. Umieszczamy w programie nazwę zmiennej i odpowiednia dla danego typu liczba komórek pamięci zostaje przydzielona na wartość zmiennej. Do wartości tej będziemy się odwoływać podając nazwę zmiennej — możemy podstawiać na zmienną nowe wartości jak też badać lub wykorzystywać w wyrażeniach wartość aktualną (ostatnio podstawioną). Generalnie korzystanie ze zmiennych nie kryje w sobie głębszej filozofii, zaś niektóre problemy związane z odpowiednim doborem typu zmiennych lub mogące wynikać z używania w jednym wyrażeniu arytmetycznym zmiennych różnych typów omawialiśmy już kilka miesięcy temu.

Gdy danych przybywa

Weźmy jako początek dalszych rozwiązań zadanie niewiarygodnie proste: wczytać i umieścić w pamięci komputera 200 liczb. Jeśli mamy do dyspozycji tylko zmienne, to musimy mieć ich dokładnie 200 i tyleż (no może kilka razy mniej) instrukcji czytania. Dla porządku zmienne te można by nazwać np. kolejno: Z1, Z2, Z3, Z4 itd., aż do Z199, Z200. Proponuję żebyście sami napisali fragment programu wyczytującego dla nich wartości.

Spróbujmy teraz podejść do problemu inaczej. Zamiast dwustu zmiennych stworzymy tablicę mającą dwieście elementów. Tablicy nadamy nazwę, np. Z, natomiast do elementów będziemy się odwoływać podając nazwę tablicy i numer elementu. Można by to zapisać np. tak: Z(1), Z(2)..., Z(199), Z(200). Można też w odwołaniach do elementów tablicy używać nie tylko stałych (1, 2, 199 itd.) lub zmiennych, np. Z(i) lub nawet wyrażeń arytmetycznych np. Z(i+j+2). Dzięki temu zapis programu wyczytującego 200 wartości skróci się do:

```
for i=l to 200 do
read Z(i)
end for i
```

Gdy zmienna i będzie przyjmować kolejne wartości 1, 2, 3 itd. wczytywane będą kolejno elementy Z(1), Z(2) itd.

Taki pojedynczy element tablicy właściwie to samo co zmienna — też może przechowywać pojedynczą wartość. Wartość tę można zmieniać oraz wykorzystywać tak samo jak wartość zmiennej. Numer elementu często nazywany jest indeksem a podobieństwo między nimi a zmiennymi prostymi. Omawiając poprzednio zmienne proste zwróciliśmy uwagę na ich typ (typ wartości, które można im nadawać). Typ wszystkich elementów tablicy jest jednakowy, określamy go definiując tablicę w skrócie mówimy też często „typ tablicy". Tak więc możemy mieć tablice całkowite — ich elementy to liczby typu całkowitego, tablice rzeczywiste (zawierające liczby typu rzeczywistego), tablice tekstowe czy logiczne.

Potrzeby dyktują rozwiązania

Tablice doskonale pasują do reprezentacji danych dla wielu problemów rozwiązywanych przez matematykę i jej rozliczne zastosowania. Jeśli zadanie zaczyna się od słów: „mamy dane n liczb, a1, a2... an..." to bez tablicy trudno sobie z nim poradzić.

W matematyce występują również problemy, które nie dają się wygodnie opisać ciągiem liczb. Np. jeśli mamy układ równań:

$$3x + 5y + z = 11$$
$$x - 6y - 2z = 88$$
$$-5x + y + 4z = 0$$

to występujące w nim liczby (współczynniki układu) wygodnie jest zapisać w postaci prostokątnej tablicy.

3	5	1	11
1	−6	−2	88
−5	1	4	8

Elementy tej tablicy wygodnie jest oznaczyć np. tak:

a11	a12	a13	a14
a21	a22	a23	a24
a31	a32	a33	a34

Do każdego z tych elementów odwołujemy się podając dwie liczby: numer wiersza i numer kolumny w której się znajduje, np. a23 oznacza liczbę z wiersza 2 i kolumny 3.

Żeby reprezentować takie obiekty w programach używamy tablic z dwoma wskaźnikami (indeksami). Czasami nazywamy je dwuwymiarowymi, a omówione poprzednio tablice z jednym wskaźnikiem jednowymiarowymi.

W większości języków programowania tablica przed użyciem musi być zadeklarowana tzn. trzeba określić jej nazwę, typ, liczbę wskaźników, oraz liczbę elementów. Liczbę elementów określamy zwykle podając najmniejszą i największą wartość wskaźnika. Np. A(1:200) określa tablicę złożoną z 200 elementów, od A(1) do A(200), TABL(1:10, 1:20) określa dwuwymiarową tablicę mającą 10 wierszy i 20 kolumn, a więc 200 elementów. Kto tego nie widzi niech sobie rozpisze elementy tej tablicy po wierszu i policzy je:

(1,1) (1,2) (1,3)..........
(2,1 (2,2)...........

W BASIC-u tablice deklarujemy poleceniem DIM, np.:

DIM WW (100)

oznacza to tablicę zawierającą 101 elementów, WW(0), WW(1) do WW(100), gdyż BASIC domyślnie zawsze przyjmuje dolną granicę zmienności wskaźnika jako 0. Tak więc DIM X(20) równoważne jest w naszej konwencji X (0:20).

Jako przykład odwoływania się do tablicy dwuwymiarowej napiszemy fragment wpisujący zero do wszystkich elementów tablicy:

```
10 DIM TB(10,10)
20 FOR I=0 TO 10
30 FOR J=0 TO 10
40 TB(I,J)=0
50 NEXT J
60 NEXT I
```

Przy okazji, ile elementów ma tablica TB z powyższego przykładu? — odpowiedź na końcu artykułu.

Uwaga na granice

Przez analogię do tablic dwuwymiarowych możemy łatwo stworzyć tablice trójwymiarowe, czyli mające trzy wskaźniki czterywymiarowe itd. w zależności od potrzeb i... możliwości języka programowania, którym się posługujemy.

Oczywiście zastosowania tablic nie ograniczają się do zadań matematycznych. Np. za miesiąc użyjemy ich rozwiązując problem wyszukiwania informacji.

Mówiąc o wykorzystywaniu tablic warto zwrócić uwagę na dość typowy błąd występujący przy odwołaniach do nich. Przy tablicy TB z poprzedniego przykładu podstawienie:

P=TB (1,30)

nie ma sensu, gdyż taki element nie istnieje w tablicy TB (odwołujemy się do elementu z kolumny 30, a tablica ma tylko 20 kolumn). Ponieważ jako indeksów używaliśmy stałych, błąd widać gołym okiem. Jeśli jednak w programie mamy np:

```
I=A+B+33
J=(I−5) 3
TB(I,J)=55
```

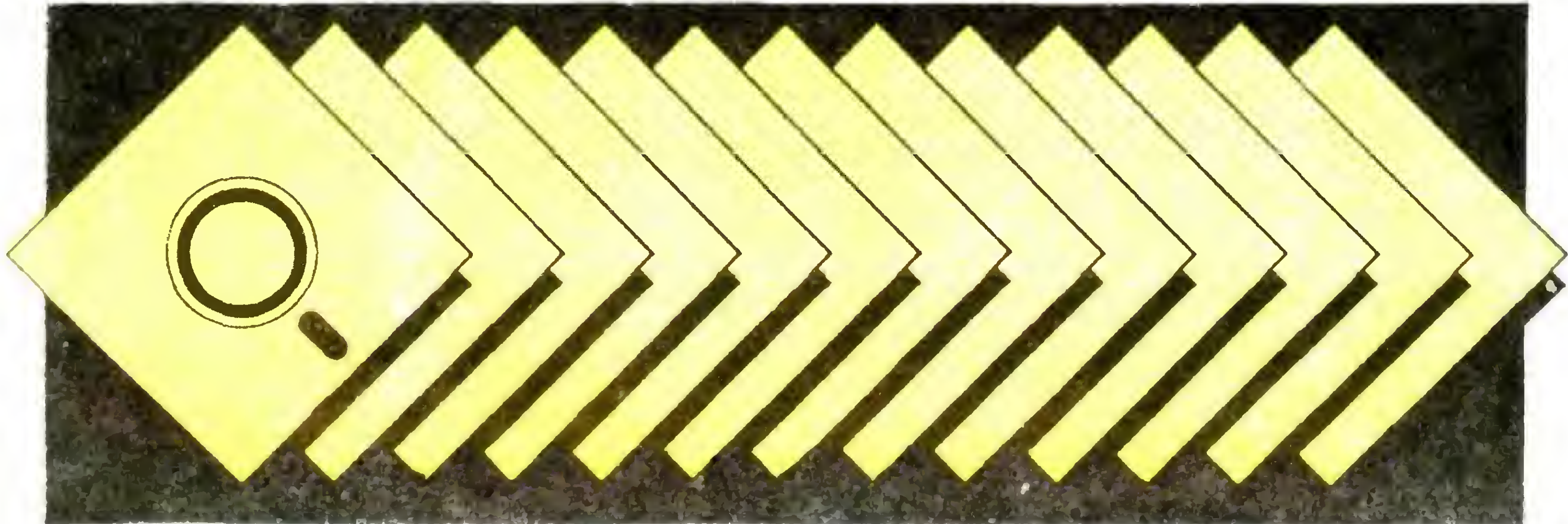
to prawidłowość odwołania zależy od wartości zmiennych w czasie pracy programu i bez szczegółowej analizy tych wartości nie jesteśmy w stanie stwierdzić, że program odwołuje się do nieistniejącego elementu tablicy. Ponieważ błędy tego typu zdarzają się dosyć często większość języków programowania daje możliwość kontroli w czasie wykonywania programu, czy aktualna wartość wskaźników mieści się w przedziale podanym przy deklaracji. Taka kontrola zmniejsza szybkość działania programu, lecz zabezpiecza przed dezorganizacją powodowaną przez odwołania do nieistniejących elementów tablicy.

Do problemów związanych z optymalnym wykorzystaniem tablic i innych struktur danych wrócimy w przyszłym miesiącu.

x) Tablica TB (10,10) czyli ze wskaźnikami z przedziału (0:10, 0:10) ma 11 11=121 elementów.

Andrzej Krul

„O ZŁOTĄ DYSKIETKĘ BAJTKA"



Przedstawiamy dzisiaj pierwsze kluby, które zadeklarowały swój udział we współzawodnictwie „O Złotą Dyskietkę BAJTKA". W kilku kolejnych numerach zaprezentujemy naszym Czytelnikom wszystkie kluby. Jury naszego konkursu wybierze spośród nich kilka klubów finałowych, które będą rywalizowały ze sobą bezpośrednio. Finał odbędzie się przed kamerami telewizji, tak więc wszyscy nasi Czytelnicy będą mogli poznać najlepszych. *(b)*

CELBIT

Kwidzyn

Klub CELBIT powstał wiosną 1986 roku w Kwidzynie, pod patronatem Zakładów Celulozowo-Papierniczych. Nie sposób pisać o klubie nie poświęciwszy przedtem kilku słów samej „Celulozie", jak nazywają zakład jego pracownicy i mieszkańcy miasta. To bardzo nowoczesny kombinat, który nie boi się porównania z fabrykami papieru budowanymi na Zachodzie. Skomplikowana technologia, niezawodne wysokowydajne maszyny, to wszystko oczywiście nie może się obejść bez komputerowego sterowania. Tak więc komputery w „Celulozie" nie są dla nikogo czymś nadzwyczajnym, z drugiej jednak strony nie wszyscy przecież pracownicy mają na co dzień do czynienia z informatyką. Im właśnie, a także ich rodzinom miał przede wszystkim służyć klub „CELBIT".

Na dobry początek Zakłady kupiły dla klubu trzy Timexy wraz z magnetofonami, niedługo potem Polbrita ze stacją dysków, a wreszcie trzy Atari 800XL z osprzętem (stacje dysków, magnetofony i drukarka).

W 1987 roku klub zrzeszał już ponad 200 członków. Stał się przy tym klubem otwartym nie tylko dla pracowników Zakładów Celulozowo-Papierniczych i ich rodzin. Zyskał także własną siedzibę w świetlicy osiedlowej na Zatorzu II. W tym czasie klub zorganizował wiele imprez otwartych — pokazy sprzętu z okazji 1 Maja i Dnia Dziecka, wakacje i ferie zimowe z komputerem.

Dzięki współpracy z redakcją miesięcznika „Komputer" klub nawiązał współpracę z Hobby Computer Club z Holandii. Materialnym efektem tej współpracy pozostał Commodore 64 przekazany przez HCC.

W marcu 1988 roku „CELBIT" przeniósł się do nowej siedziby w Miejskim Ośrodku Kultury. W kilka miesięcy później stan posiadania klubu wzbogacił się o komputer IBM z drukarką Star NL 10. Służy on najbardziej zaawansowanym klubowiczom. Obecnie klub zajmuje dwa pomieszczenia. W pierwszym znajdują się trzy Atari i Timex. Na jeden dzień w tygodniu pokój ten zmienia się w salon gier komputerowych dla dzieci z całego Kwidzyna, przez resztę dni spotykają się w nim programiści amatorzy. W drugim pomieszczeniu stoi duma klubu — IBM XT, Polbrit i Commodore 64. Jest to miejsce pracy klubowiczów najbardziej zaawansowanych w programowaniu.

Adam Kubic, prezes klubu i jeden z inicjatorów jego powstania widzi potrzebę wymiany doświadczeń pomiędzy istniejącymi klubami. Brak informacji powoduje, że powstające kluby „wyważają otwarte drzwi" wymyślając od nowa to, co inni robią już od kilku lat. Takiej wymianie doświadczeń służyłby najlepiej periodyczny biuletyn informacyjny klubów komputerowych. „CELBIT" deklaruje chęć bezpośredniej redakcji takiego wydawnictwa przy współpracy innych klubów z całej Polski.

Klub komputerowy „CELBIT", ul. Warszawska 18, 82-500 Kwidzyn.

ZŁOTY AMSTRAD

Złoty Stok

Złoty Stok, przepiękne miasteczko położone w Kotlinie Kłodzkiej tuż przy samej granicy z Czechosłowacją zawsze słynęło z bogactwa. Dawniej o zamożności miasta stanowiło złoto, które eksploatowano w miejscowej kopalni aż do 1962 roku. Dzisiejsze bogactwo Złotego Stoku to wspaniałe krajobrazy, piękna architektura i — a właściwie przede wszystkim — ludzie, którzy czują się związani z tym miastem i chcą coś dla niego zrobić.

Kilku takich właśnie zapaleńców związanych z Zakładami Tworzyw i Farb w Złotym Stoku założyło w lutym 1987 roku klub komputerowy „Złoty Amstrad". Od samego początku jest to klub zakładowy, lecz jego drzwi stoją otworem przed każdym chętnym. Głównym przedmiotem zainteresowania klubowiczów są komputery Amstrad serii CPC. W zeszłym roku klub wzbogacił się o zgodnego z IBM Amstrada PC 1512.

Niedawno klub przeniósł się do nowego, nareszcie własnego pomieszczenia. Jest ono doskonałą ilustracją powiedzenia „ciasne ale własne". Kilkanaście metrów kwadratowych to niezbyt wiele, dobrze zagospodarowane, pozwoliło jednak na urządzenie kilku stanowisk pracy.

Klub współpracuje z dwiema szkołami podstawowymi ze Złotego Stoku. Uczniowie poznają na lekcjach zasady posługiwania się komputerem i korzystania z — na razie nielicznych — programów edukacyjnych.

Trzy razy w tygodniu część klubowych komputerów udostępniana jest dzieciom ze szkół, a jeden raz wszystkim chętnym. Dzieci szkolne korzystają z komputerów według listy sporządzonej przez nauczycieli w szkole. Dzięki temu wszystkie dzieci mają równe szanse, a wpisanie na listę stanowi dla nich większą motywację do nauki niż piątki w dzienniczku.

Zajęcia klubowe odbywają się w dwóch sekcjach: sekcji programowania, będącej głównym kierunkiem działania; oraz sekcji gier komputerowych, przeznaczonej dla najmłodszych dzieci i dla tych, którzy dopiero rozpoczynają komputerową edukację. W ramach sekcji programowania prowadzone są zajęcie poświęcone językom LOGO i Turbo-PASCAL, pisaniu programów przy pomocy dBase. Najbardziej zaawansowani mają możliwość poznania i tworzenia oprogramowania działającego w sieci komputerowej pracującej w zakładzie.

Klub zrzesza 130 stałych członków. Część z nich to członkowie-korespondenci. Dzięki nim właśnie, a także za sprawą uczestnictwa „Złotego Amstrada" w pierwszej edycji współzawodnictwa „O Złotą Dyskietkę BAJTKA" klub zyskał popularność w całej Polsce a nawet poza jej granicami (szczególnie w Czechosłowacji). Każdy posiadacz czy miłośnik Amstrada, który napisze do klubu otrzyma pełną listę użytkowników tego sprzętu, o których klub ma informację. Członkowie klubu służą w miarę swych możliwości pomocą merytoryczną, o ile jest to możliwe listownie. Wszystkich swych korespondentów zapraszają do klubu. Zapewniają wrażenia — z uwagi na urocze położenie Złotego Stoku — nie tylko komputerowe.

Klub ma zamiar rozpocząć również działalność wydawniczą. W planach jest gazetka klubowa i biuletyn szkoleniowy. Wydawnictwa te mają być wysyłane bezpłatnie wszystkim członkom klubu. Różne formy szkolenia zostały ostatnio wzbogacone o zorganizowanie klubu filmów popularno-naukowych wyświetlanych na video.

Coraz większa liczba zagranicznych korespondentów klubu nasunęła Zdzisławowi Kryczce i Krzysztofowi Wolińskiemu (prezesowi i opiekunowi klubu) pomysł rozszerzenia działalności na sąsiednie kraje — Związek Radziecki, Czechosłowację, Węgry i Bułgarię — kraje, w których komputery Amstrad zyskały sobie już pewną popularność. Byłby to pierwszy w Polsce tego rodzaju klub o charakterze międzynarodowym.

Klub komputerowy „Złoty Amstrad", Rynek 1, 57-250 Złoty Stok.

„HARCBAJT"

Gdańsk

Harcerski Klub Komputerowy „HarcBajt" mieści się w Domu Harcerza na gdańskiej starówce. Choć w nazwie klub posiada przymiotnik „harcerski", to przynależność do tej organizacji nie jest warunkiem koniecznym zapisania się do „HarcBajtu".

Trzon klubu stanowi grupa około 10 osób, z których każda interesuje się innymi zagadnieniami i organizuje pracę w swojej dziedzinie. Jeden dzień w tygodniu przeznaczony jest na gry komputerowe. Tego dnia klub otwarty jest dla wszystkich chętnych.

Wymiernymi efektami działania klubu jest kilka programów napisanych samodzielnie, oraz kilka przystosowanych do korzystania z polskiego alfabetu, a także proste interface'y pozwalające na współpracę różnych komputerów pomiędzy sobą i z urządzeniami peryferyjnymi.

Do tradycji działania klubu należą tak zwane „Lotne Patrole Komputerowe" odwiedzające zimowiska, obozy i domy kultury na terenie województwa gdańskiego. Wyjazdy te mają na celu przybliżenie możliwie największej liczbie młodzieży, czym jest komputer i do czego może on służyć, a także jest to propozycja rozrywki w postaci gier komputerowych.

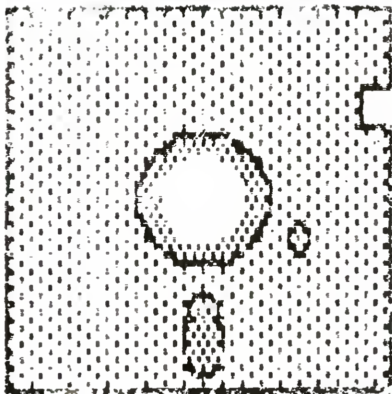
Członkowie klubu interesują się nie tylko komputerami. Pasjonuje ich wszystko co nowe i nowoczesne. Są właśnie w trakcie instalowania anteny do odbioru telewizji satelitarnej, a podjęli się także wykonania studia TV dla Domu Harcerza.

Obok klubu komputerowego w gdańskim Domu Harcerza działa także Harcerski Klub Fantastyki „KROMLECH". Tak się składa, że członkami obydwóch tych klubów są w znacznej większości te same osoby. Siłą rzeczy działalność „KROMLECHA" i „HarcBajtu" wiele łączy ze sobą. Takim właśnie wspólnym przedsięwzięciem ma być wydawanie klubowego periodyku poświęconego fantastyce, a w następnej kolejności drugiego, poświęconego komputerom.

Harcerski Klub Komputerowy „HarcBajt", ul. Za Murami 2—10, 80-823 Gdańsk.

GIEŁDA NA DZIEŃ 14.04.1989

	Giełda Bajtka	Sklep Bajtka	Komis	Pewex	RFN	Wlk. Brytania	CSH i inne
	tys. zł	tys. zł	tys. zł	$	DM	#	tys. zł
SINCLAIR							
ZX 81	90	80	—	—	—	—	—
ZX Spectrum 48	330	280	350	115	80	40	—
ZX Spectrum +	360	350	400	—	90	55	—
Timex 2048	320	330	400	146	—	—	240
ZX Spectrum 128+	430	—	—	—	—	85	—
ZX Spectrum 128+2	—	—	—	—	140	100	—
ZX Spectrum 128+3	—	—	—	—	280	150	—
drukarka Seikosha GP 50s	—	220	—	—	—	—	—
Interface Kempston	15	10–15	—	—	10	9	9.3
COMMODORE							
Commodore 64	600	650	710	219	290	120	—
VC 20	100	180	—	—	—	—	—
C 16	180	250	—	—	80	—	—
C 116	220	240	—	—	70	40	—
C Plus 4	310	360	380	—	150	—	—
C 128	900	910	—	299	399	170	—
C 128 D	1700	—	—	—	820	250	—
Amiga 500	2200	2900	—	—	899	280	—
Magnetofon 1531	100	120	170	48	30	49	—
Stacja dysków 1541	530	680	—	—	355	—	—
Stacja dysków 1571	780	800	—	299	460	160	—
Drukarka MPS 803	500	600	—	—	260	100	—
ATARI							
Atari 800 XL	380	500	470	—	—	—	—
Atari 65 XE	400	510	520	114	100	80	—
Atari 130 XE	500	680	—	199	220	110	—
Atari 520 ST	1500	1800	—	—	—	260	—
Atari 1040 ST	—	—	—	—	1140	499	—
Magnetofon XC 12	120	130	—	36	40	42	—
Stacja dysków 1050	700	700	750	185	300	100	—
Stacja dysków 520 STM	—	500	—	—	—	120	—
Drukarka 1029	400	650	—	—	—	—	—
AMSTRAD							
Amstrad 464 mono.	700	700	—	—	350	150	—
Amstrad 664 mono.	—	900	—	—	—	—	—
Amstrad 6128 mono.	110	—	—	—	670	220	—
Amstrad PCW 8256	—	—	—	—	—	300	—
Amstrad PCW 8512	—	—	—	—	—	390	—
Amstrad PCW 9512	—	—	—	—	—	450	—
Stacja dysków do 464	—	—	—	—	380	100	—
SHARP							
Sharp MZ 700	—	360	—	—	—	—	—
Sharp MZ 800	—	420	—	—	—	—	—
Dyskietki 5.25 cala	2	2-2.5	3	1	0.7	0.5	4-9
Dyskietki 3.5 cala	7.5	7.5-9	7-9	—	5	1-2	9-10
Dyskietki 3 cale	11	—	—	—	6	2	10
Joystick	21	16-26	20-30	5	10	5	14
Monitor Neptun	60	80	—	—	—	—	68



INDYWIDUALNY BANK DANYCH

Jarosław Zgódka, lat 17. Posiada mikrokomputer C-64, magnetofon, joystick. Oprogramowanie: gry, programy muzyczne i użytkowe. Zainteresowania: mikrokomputery, sport, muzyka. Pragnie nawiązać kontakt w celu wymiany doświadczeń oraz oprogramowania. Adres: 51-639 Wrocław, ul. Wojtkiewicza 7/1.

Piotr Kolanek, uczeń 17 lat. Posiada Atari 130 XE, stację dysków LDW Super 2000 oraz magnetofon XC 12. Dysponuje zestawem około 150 gier oraz 50 programów użytkowych. Zainteresowany muzyką i grafiką komputerową. Pragnie nawiązać kontakt z ludźmi o podobnych zainteresowaniach w celu wymiany programów. Adres: 62-800 Kalisz, ul. Polna 31/29.

Władysław Krupa, 58-410 Marciszów, ul. Kościelna 1/3 odstąpi następujące numery „Bajtka": 1—4/85, 1—12/86, 1—7, 9—10/87.

Marcin Szczygielski, lat 16, uczeń LO. Posiada AMSTRADA CPC 6128 k, dwa joysticki, oraz około 100 gier i programów użytkowych. Zainteresowania: informatyka, sport, język niemiecki. Chciałby nawiązać kontakt z innymi posiadaczami tego mikrokomputera. Adres: 68-200 Żary, ul. 1 Maja 10/11/III k.

Ireneusz Pietrzak, posiada Atari 65XE, magnetofon. Pragnie nawiązać kontakt z posiadaczami tego komputera w celu wymiany oprogramowania. Adres: 98-300 Wieluń, skr. poczt. 51.

Darek Gołębiewski, lat 15. Posiada mikrokomputer APPLE II, stację dysków, monitor oraz joystick. Oprogramowanie: gry i programy użytkowe. Prosi o kontakt posiadaczy tego mikrokomputera w celu wymiany oprogramowania. Adres: 53-148 Wrocław, ul. Jarzębia 9/73

Grzegorz Żurakowski prosi o kontakt posiadaczy komputera COMMODORE 20 VIC. Adres: Wrocław, ul. Grottgera 16, tel. 48-12-34.

Adam Krzyżanowski, lat 20. Posiada COMMODORE 64, magnetofon 1531, mysz. Oprogramowanie: programy muzyczne, gry oraz programy użytkowe. Pragnie nawiązać kontakt w celu wymiany programów i doświadczeń. Adres: 84-100 Puck, ul. Przebendowskiego 19/45 c.

Ulf Gebhard, posiada mikrokomputer C-64. Pragnie nawiązać kontakt w celu wymiany oprogramowania, doświadczeń oraz literatury. Korespondencja w języku niemieckim. Adres: R.-Rothkegel-str.47, 7500 Cottbus, DDR.

Jacek Kamiński, student, lat 23. Posiada Commodore 116 z magnetofonem 1531. Nawiąże korespondencję w celu wymiany doświadczeń. Prosi o pomoc w zdobyciu programów. Interesuje się językiem maszynowym. Adres: Mława, ul. Górna 25.

Rafał Szczepkowski, lat 10. Posiada mikrokomputer Timex 2048. Pragnie nawiązać kontakt z posiadaczami tego komputera w celu wymiany oprogramowania i doświadczeń. Adres: 80-441 Gdańsk, ul. Dzierżyńskiego 11/10.

Andrzej Załuski, lat 18. Posiada mikrokomputer Commodore VC-20, magnetofon VC-1530 i rozszerzenie pamięci + 16 kB. Oprogramowanie: programy użytkowe, muzyczne, edukacyjne oraz gry. Adres: 73-300 Łobez, ul. Murarska 9/11.

Paweł Burakowski, lat 15. Posiada ZX Spectrum Plus. Oprogramowanie: około 300 gier i programów użytkowych. Zainteresowany jest wszystkim co jest związane ze światem komputerów. Adres: 32-300 Olkusz, ul. Sławkowska 5/37



Pascal, który na początku swej kariery był mało znanym językiem stosowanym raczej do celów dydaktycznych niż do pisania poważnych programów, doczekał się w momencie wprowadzenia na rynek przez firmę Borland kompilatora Turbo Pascal swego rozkwitu. Dziś jest jednym z najpopularniejszych języków programowania, zwłaszcza wśród użytkowników sprzętu mikro. W 1987 roku na rynku pojawiła się nowa wersja tego kompilatora, obdarzona numerem 4.0, i w ciągu kilku tygodni stała się szlagierem. Wprawdzie dzięki doskonale opracowanej opcji Help doświadczony programista może spokojnie posługiwać się nowym Turbo Pascalem, jednak by wykorzystać wszystkie jego możliwości i niuanse potrzebna jest jakaś dodatkowa pomoc, której udzieliły w zeszłym roku WKiŁ, wydając książkę „Rozszerzony Turbo Pascal wersja 4.0" Jana Bieleckiego.

Książka składa się z trzech części. Pierwsza z nich to opis tak zwanego zintegrowanego środowiska operacyjnego, czyli połączonego w jedną całość edytora i kompilatora. Znalazł się tu dokładny przewodnik po wszystkich możliwych opcjach dostępnych dla użytkownika podczas pisania i uruchamiania programu, oraz kilka uwag o edytorze, którego dokładny opis znajduje się w dodatku B. Druga część książki to opis języka — jego składni i predefiniowanych typów, funkcji i procedur. Opis jest precyzyjny i wyczerpujący, toteż można z niego korzystać by rozwiać wątpliwości jakie pojawiają się czasem podczas pisania programów. Nie radziłbym jednak uczyć się na jego podstawie Pascala — na to jest zbyt suchy, i niestety odbiega od przyjętych w naszej literaturze standardów. Trzecia wreszcie część, mająca największą wartość dla wszystkich znających w miarę dobrze język, to opis bardzo szerokiej bioblioteki procedur i funkcji, zgrupowanych w sześciu tak zwanych modułach, będących największą zaletą nowej wersji Pascala. Same moduły, ich koncepcja i sposób posługiwania się nimi zostały omówione wcześniej, razem z opisem składni języka. W trzeciej części książki opisane są dokładnie wszystkie procedury, stałe i zmienne dostępne dla użytkownika modułów rozprowadzanych przez Borland razem z kompilatorem. Każda procedura ma dokładnie opisaną składnię, wykonywane czynności oraz uwagi na temat działania i istniejących ograniczeń. Na końcu książki znajduje się kilka dodatków, zawierających różne informacje, które mogą okazać się użyteczne podczas pracy.

Książka dla wielu użytkowników Turbo Pascala 4.0, nie mających dostępu do oryginalnej instrukcji, stanowi poważną pomoc. Tym bardziej szkoda, że znajdujący się w niej skorowidz jest tak ubogi. Nie ma w nim wielu haseł, które moim zdaniem znaleźć się powinny, np. dyrektyw IFDEF, ENDIF itd., trzeba ich szukać pod „dyrektywy kompilacji warunkowej". Nie ma także (i to w całej książce, nie tylko w skorowidzu) listy podprogramów wchodzących w skład modułów. Oczywiście wszystkie są opisane, ale brak zwykłej alfabetycznej listy, która mogłaby bardzo ułatwić życie użytkownika. Brak również opisu sposobu poprawiania programów napisanych we wcześniejszych wersjach Turbo Pascala tak, by mogły być skompilowane przy użyciu nowego, znacznie lepszego od poprzednich wersji kompilatora. Te informacje mogłyby znaleźć się w książce zamiast niektórych przykładów, bardzo przegadanych i niepotrzebnych dla zrozumienia co robią procedury takie jak CIRCLE, rysująca kółko.

Ostatnie zastrzeżenie jest już innego rodzaju, i dotyczy języka jakim książka została napisana. Przyjęta przez autora konwencja wprowadzania angielskich nazw zmiennych w przykładach, dała na stronie 186 efekt w postaci następującego zdania: „Usunięcie z source, podciągu liczącego count znaków zaczynającego się od znaku o numerze index". Czy nie lepiej napisać to po polsku?

Wymienione wady (nie wszystkie które zwróciły moją uwagę podczas czytania), nie dyskwalifikują wcale książki jako przydatnego narzędzia. Dla wielu użytkowników komputerów, zwłaszcza tych piszących oprogramowanie na własne potrzeby, będzie ona na pewno jedną z podstawowych pozycji leżących obok komputera, i bardzo często i intensywnie używanych. Drugą taką pozycją jest opisana już w Bajtku książka „IBM PC i PC DOS", również wydana przez WKiŁ, za co składam wydawnictwu serdeczne podziękowanie w imieniu tych wszystkich, którym udało się obie (albo przynajmniej jedną) z tych książek kupić. Ciekawe, kiedy możemy się spodziewać opisu Turbo Pascala 5.0, który (piszę te słowa pod koniec grudnia) dotarł już do Polski?

(mb)

Jan Bielecki, „Rozszerzony Turbo Pascal wersja 4.0", Wydawnictwa Komunikacji i Łączności, Warszawa 1988, wydanie I, nakład 30 000, cena 1200 zł.

IMIENINY CIOCI KLOCI

Cześć Maluchy!

Wczoraj były imieniny ukochanej cioci Kubusia, Cioci Kloci, a Kuba nawet nie wysłał jej kartki z życzeniami. A przecież ona zawsze pamięta o każdym święcie. Cóż robić, Kubuś wyśle dzisiaj kartkę z przeprosinami, ale swoją drogą przydałoby się zaradzić temu na przyszłość. Ale jak?

Do dzisiaj medycyna nie znalazła niestety skutecznego medykamentu poprawiającego pamięć. Tak więc nie szukajmy ratunku w pigułkach, maściach i rozmaitych miksturach. Może więc lepiej zanotować wszystko w kalendarzu? Ale przecież w kalendarzu Kubusia imieniny Cioci Kloci zakreślone są grubym, czerwonym flamastrem, a on i tak o nich zapomniał. Stosunkowo najlepiej do przypominania nadaje się mama. Ona wszystko doskonale pamięta. Mama jednak nie przypomni o swoich imieninach, Dniu Kobiet czy Dniu Matki.

Kogo by więc poprosić o tę przysługę? Ten ktoś musi mieć dobrą pamięć i dużo cierpliwości. Kubuś dobrze wie, kto ma najlepszą pamięć i nigdy się nie znudzi. To oczywiście komputer. Spróbujmy teraz, razem z Kubusiem nauczyć komputer przypominania nam o tym, o czym sami możemy zapomnieć.

Popatrzmy jaki program napisał Kubuś i prześledźmy jego działanie. Pierwsza część programu (linie 100—180) umożliwia podanie komputerowi aktualnej daty. Komputer pyta najpierw o dzień (linie 130 i 140), a następnie o miesiąc (linie 150 i 160). Rok nie jest mu potrzebny, ponieważ program w wersji napisanej przez Kubusia przypominać ma jedynie o terminach stałych, takich jak urodziny i imieniny.

Komputer nie przyjmie odpowiedzi, które z założenia są błędne, np. nie przyjmie liczby 13 jako numeru miesiąca. Dzieje się tak, dzięki liniom 145 i 165, w których sprawdzane jest, czy wpisana przez użytkownika liczba jest całkowita i czy nie wykracza poza dozwolony przedział. Dla zmiennej dzień wynosi on ‹1—31›, dla zmiennej miesiąc —‹1—12›.

Teraz komputer przechodzi do porównywania daty podanej jako dzisiejsza z terminami przechowywanymi w pamięci, a konkretnie w liniach DATA (linie 1000—5000) na końcu programu. Informacje w liniach danych zapisywane są w ściśle ustalonej kolejności. Najpierw dwie zmienne liczbowe, dzień i miesiąc, a następnie tekst określający czego dotyczy zapamiętany termin. Ostatnia wartość wynosi 100 i oznacza koniec zapisu, a dla komputera koniec odczytywania danych.

Komputer wczytuje dane z jednej linii (dotyczące jednego terminu) (linie 200—230). Warunek w linii 210 umożliwia wyskoczenie z pętli po wczytaniu wszystkich danych. Następnie termin porównywany jest z dzisiejszą datą (linie 240—270). Jeśli do wyznaczonego terminu pozostało dziesięć lub mniej dni informacja o nim ukazuje się na ekranie.

Zapewne zauważyliście, że terminy obliczane są tak, jak gdyby wszystkie miesiące miały po 31 dni. Nie powoduje to jednak większego błędu, gdyż dla nas ma znaczenie liczba dni pomiędzy stosunkowo bliskimi terminami a nie liczba dni, które upłynęły od początku roku. Tak więc błąd może wynosić 1 dzień w przypadku miesięcy 30 dniowych i 3 dni w lutym, jeśli ma on 28 dni. Błąd ten powstaje pod koniec miesięcy „krótszych", gdy i opóźnia komunikat o 1—3 dni.

Program można uzupełnić o dodatkowe funkcje. Może on na przykład informować ile lat kończy nasz solenizant (choć nie jestem pewny, czy wszystkie panie byłyby zadowolone). W każdym razie linie danych należy wypełnić datami imienin, urodzin i rocznic naszych znajomych i uruchamiać ten program chociaż dwa razy w tygodniu. Jeśli tak zrobisz, z pewnością nie zapomnisz o żadnym z nich. A może także nie zapomnisz o przyjaciołach Kubusia?

Romek

```
100 PRINT
110 PRINT "***** Przypominacz *****"
120 PRINT
130 PRINT "Podaj dzien"
140 INPUT dzien
145 IF dzien>31 OR dzien<1 OR dzien<>INT
(dzien) THEN GOTO 130
150 PRINT "Podaj miesiac cyfra"
160 INPUT miesiac
165 IF miesiac>12 OR miesiac<1 OR miesia
c<>INT(miesiac) THEN GOTO 150
170 PRINT
180 PRINT
200 READ xdzien
210 IF xdzien=100 THEN GOTO 500
220 READ xmiesiac
230 READ tekst$
240 LET l1=(xmiesiac-1)*31+xdzien
250 LET l2=(miesiac-1)*31+dzien
260 LET l=l1-l2
270 IF l>=0 AND l<=10 THEN PRINT xdzien;
xmiesiac;tekst$
280 GOTO 200
500 PRINT
510 PRINT "To wszystko, dziekuje."
520 PRINT "Nie zapomnij wlaczyc mnie jut
ro."
1000 DATA 26, 5, "Dzien Matki"
1010 DATA 21, 1, "Dzien Babci"
1020 DATA 8, 3, "Miedzynarodowy Dzie
Kobiet"
1030 DATA 23, 6, "Dzien Ojca"
1040 DATA 30, 9, "Dzien Chlopaka"
1050 DATA 14,10, "Dzien Nauczyciela"
1060 DATA 1, 6, "Dzien Dziecka"
1070 DATA 6,12, "Sw. Mikolaja"
1080 DATA 15, 1, "Urodziny Kubusia"
1090 DATA 22, 7, "Urodziny Magdalenki"
1100 DATA 12, 6, "Urodziny Zuzanki"
1110 DATA 22, 3, "Urodziny Romka"
5000 DATA 100
10000 REM W przypadku Atari nalezy
dopisac linie 10 dim tekst$(30) i usunac
 wszystkie cudzyslowy z linii DATA
```


Na listy Czytelników odpowiada Dominik Falkowski

Drogi Bajtku!

Do napisania tego listu skłoniły mnie dwie sprawy:
Po pierwsze chciałbym się dowiedzieć w jakiej cenie są ogłoszenia w Indywidualnym Banku Danych oraz w rubryce SOS.
Po drugie chcę napisać kilka słów na temat firmy Pana Andrzeja Zakutyńskiego „Progres" działającej w Koszalinie. Otóż po wcześniejszym listownym umówieniu się z wyżej wymienioną firmą w dniu 9 stycznia 1989 wysłałem kasetę razem z 12 programami na wymianę. Gdy po upływie 3 tygodni od wysłania paczka nie przychodziła, wysłałem ekspresem list z prośbą o szybki zwrot kasety. Odpowiedzi jednak na ten list nie otrzymałem. Gdy piszę ten list jest 20 luty 1989, więc minął już dawno miesiąc, a ja wciąż nie otrzymuję paczki. W katalogu gier firmy 6 punkt brzmi następująco „Termin realizacji zamówień tylko 12 godzin !!!". Wydaje mi się, że z tym punktem mocno przesadzono. Niech list ten stanie się przestrogą dla tych wszystkich, którzy zamierzają skorzystać z usług „Progresu".

Andrzej Józefczuk
Lublin

Ogłoszenia zamieszczane we wspomnianych rubrykach są całkowicie bezpłatne. Są to miejsca w Bajtku przeznaczone dla tych, którzy pomagają nam redagować pismo, a więc dla Was wszystkich.
W drugiej sprawie osądzać nie będziemy. Zamieszczamy ten list jako ostrzeżenie Czytelnika dla Czytelników.

Czy magnetofon MK 232P nadaje się do współpracy z Commodore C16/116?
Czy myszka i pióro świetlne mogą być przyłączane do tych komputerów?
Nazwisko i adres do wiadomości redakcji

Jak wszystkie komputery firmy Commodore tak i C16/116 wymagają do współpracy specjalnych, firmowych magnetofonów. Są oczywiście elektronicy, którzy we własnym zakresie potrafią dokonać odpowiednich przeróbek, nie jest to jednak najprostsza sprawa nawet dla specjalisty.
W systemie operacyjnym tych komputerów nie przewidziano możliwości obsługi ani myszy, ani pióra świetlnego czy też modemu. Nie ma więc odpowiedniej obsługi portów. Nie istnieje *User port* niezbędny np. do podłączenia do niego modemu. Istnieje wszakże na rynku pióro świetlne specjalnie przystosowane do współpracy z C116.

Słyszałem, że istnieje możliwość kupna dekodera, dzięki któremu na PLUS/4 mógłbym skorzystać z programów pisanych dla C-64.

Hubert Lubecki
ul. Mickiewicza 1 m 17
19-400 Olecko



Dekoder taki nie istnieje. W komputerach C16/116 jak i PLUS/4 funkcje kontrolera dźwięku i obrazu spełnia jeden jedyny układ znany pod nazwą TED. W komputerze C-64 funkcje te są jak wiadomo rozdzielone pomiędzy dwa wysoko wyspecjalizowane układy VIC (zajmujący się grafiką) i SIŁD (zajmujący się dźwiękiem). Ponadto podczas pracy C+4 intensywnie bankuje pamięć, podczas gdy w C-64 mechanizm ten nie jest stosowany (przynajmniej teoretycznie).

Jaki telewizor może współpracować z komputerami Commodore C116 i PLUS/4, a także z ATARI 65XE?
Jaki joystick należy kupić do PULS/4?
Nazwisko i adres do wiadomości redakcji

Wszystkie komputery wyposażone w modulator TV przystosowane są do współpracy z telewizorami posiadającymi wejście antenowe 75Ω. Nawet gdy wartość ta jest inna (np. 240Ω dla starych niemieckich telewizorów), to potrzebny jest jedynie odpowiedni symetryzator dopasowujący impedancję falową. Jest to jednak przypadek niesłychanie rzadki, gdyż większość telewizorów posiada już wspomniane wejścia 75Ω.
Od czasu do czasu można na różnego rodzaju wyprzedażach nabyć specjalne joysticki do PLUS/4. Najwygodniej jest jednak kupić dowolny joystick np. Quickshot z 9-bolcowym gniazdem i zastosować złącze pośrednie. Takie złącza występowały kiedyś wraz z pakietem oprogramowania (3 kasety) przeznaczonego dla C116.

Panie Marcinie (list pisany był jeszcze do mojego poprzednika)!
Jeżeli uruchomimy ten dołączony programik i wpalcujemy różne literki, znaczki lub cyferki — otrzymamy wyniki zgodne z oczekiwaniem. Za wyjątkiem jednego przecinka!! Po jego wprowadzeniu ujawnia się komunikat REDO FROM START. A znak przecinka wg kodu ASCII ma numer 44, a więc jest niższy od 48. Dlaczego?
Krzysztof Czerwiński
ul. Winiarskiego 32/21
30-836 Kraków

```
10 INPUT"", A$
20 B1=0
30 FOR I=1 TO LEN (A$)
40 IF ASC (MID$ (A$,I,1)) <48 OR ASC (MID$
(A$,I,1))>57 THEN B1=1
50 NEXT
60 IF B1=1 THEN PRINT "WARTOŚĆ NIENU-
MERYCZNA" ELSE
   PRINT "WARTOŚĆ NUMERYCZNA"
```

Jak wiadomo, za pomocą jednej instrukcji INPUT możemy wprowadzić do komputera szereg danych. W takim wypadku piszemy np. INPUT a,b,c itd. Jak widać w instrukcji INPUT przecinek spełnia rolę specjalnego separatora wprowadzanych danych. Jest to więc znak zastrzeżony i traktowany przez komputer nie jako dana, lecz jako oznaczenie, że komputer spodziewa się kolejnych danych.

Jestem posiadaczem Commodore C-128D. Jak wiadomo ma on wbudowaną stację dysków 1571, ale nie spełnia ona w pełni moich wymagań. Potrzebuję o wiele szybszej stacji o większej pojemności. W numerze 9/88 dowiedziałem się, że istnieje stacja 1581. Proszę o jakiekolwiek informacje o niej.
Nazwisko i adres do wiadomości redakcji

Z wyników ankiet wynika, że 75% Czytelników czyta Bajtka od pierwszej do ostatniej strony a 25% tylko swój klan. Tak czy owak użytkownik Commodore powinien raczej czytać swój klan, wtedy bowiem znalazłby odpowiedź na to i na inne pytania.
Odsyłam więc do numeru „Tylko o Commodore", w którym zamieszczony był opis stacji 1581. Jako uzupełnienie mogę dodać, że jest to dobra stacja, ale niestety nie istnieją jeszcze na nią dobre programy kopiujące. Tak więc warto się zastanowić nad alternatywnym rozwiązaniem jakim są różnego rodzaju przyspieszacze, np. moduły DOLPHIN DOS, WARP i inne. Są też programy przyspieszające wczytywanie programów z dysku, ale ich wadą jest to, że nie działają one ze wszystkimi programami istniejącymi na rynku.

OKNO NA ŚWIAT

Dokończenie ze str. 32

obejmuje ponad 60.000 pozycji. Kupujący może zapytać o aktualną cenę, rabaty, no i oczywiście zamówić dowolny artykuł.

Transakcje finansowe (GO BANKING). System ten ułatwia zawieranie transakcji bankowych, wymianę poczty elektronicznej z bankiem i uzyskanie wszelkich informacji. Za pośrednictwem komputera można między innymi dokonywać kupna i sprzedaży papierów wartościowych.

Gry (GO GAMES). Twórcy CompuServe nie zapomnieli o rozrywkach. Każdy chętny może stać się uczestnikiem jednej z wielu gier przygodowych, sportowych, wojennych lub science fiction. W niektórych grach zmaga się jednocześnie kilkunastu partnerów z odległych zakątków kraju. Rywalizacja jest wtedy o wiele ciekawsza niż w grze komputerowej rozgrywanej w tradycyjny sposób.

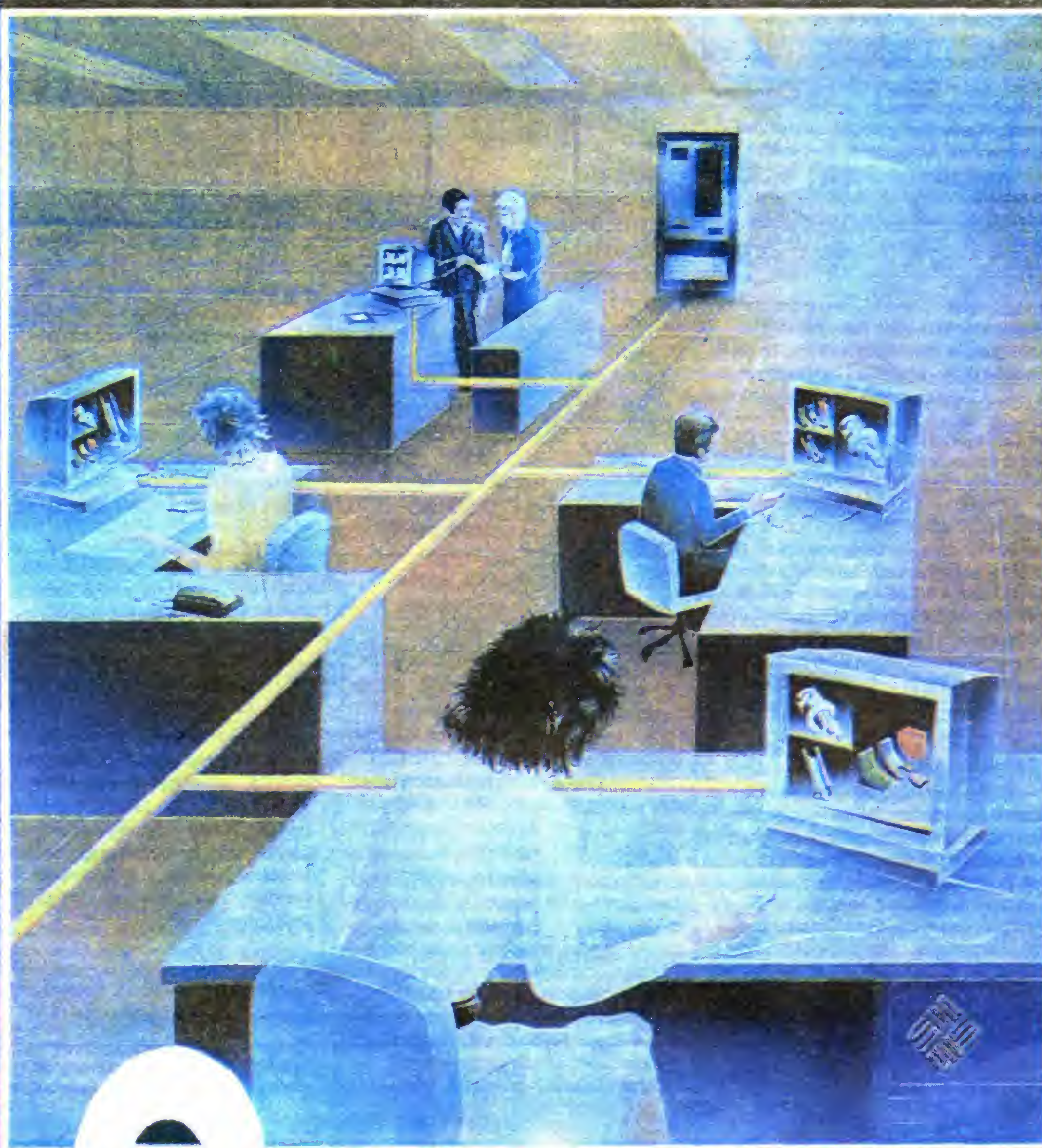
Podróże (GO TRAVEL). Usługi sieci są cenną pomocą dla milionów ludzi podróżujących służbowo lub turystycznie. CompuServe ułatwia zaplanowanie trasy i załatwienie wielu formalności, na które zwykle traci się wiele godzin. Korzystając z sieci możemy samodzielnie dokonać rezerwacji i zakupu biletu lotniczego na dowolny samolot jednej z 250 linii lotniczych. CompuServe ma bowiem bezpośrednie połączenie z największym elektronicznym bankiem danych (Official Airline Guide) zawierającym informacje o ponad 2.000.000 bezpośrednich i kombinowanych połączeniach lotniczych na całym świecie. Na życzenie komputer poda również, czy są wolne miejsca w hotelu w punkcie docelowym naszej podróży. Nocleg lub inne usługi rezerwować można w jednym z 27.000 hoteli. Podróżnych interesują nie tylko informacje o hotelach i połączeniach lotniczych. Wybierający się za granicę może zapytać o aktualne kursy walut, przepisy celne i paszportowe obowiązujące w innych krajach, wymagane szczepienia, a nawet o sytuację polityczną.

Wszystko dla domu (GO HOME). Każda rodzina znajdzie tu cenną pomoc w prowadzeniu gospodarstwa domowego. Można tu uzyskać porady prawne, finansowe i ubezpieczeniowe. Gospodyni może skorzystać z elektronicznej książki kucharskiej, w której znajdują się setki przepisów na różne dania. Komputer chętnie zastępuje także domowego lekarza. Odpowiada on na pytanie dotyczące symptomów wielu chorób i udziela porad medycznych.

Nauka (GO REFERENCE). Ta forma zainteresuje wielu studentów, którzy zbierają materiały do prac naukowych i egzaminów. Znajdują się tu informacje o aktualnie ukazujących się publikacjach. Można również skorzystać z ponad 700 banków danych z różnych dziedzin wiedzy.

Jedną z najatrakcyjniejszych usług oferowanych przez CompuServe jest poczta elektroniczna. Dzięki elektronicznej skrzynce pocztowej możemy prowadzić korespondencję z innymi użytkownikami sieci. Jeśli chcemy rozwiązać jakiś trudny problem, to wystarczy rozesłać elektroniczne SOS do osób o podobnych zainteresowaniach. Odpowiedź nadejdzie na pewno bardzo szybko.

Janusz Jarmoch




OKNO NA ŚWIAT

Twój komputer jest nie tylko cierpliwym partnerem do gry w brydża i niezawodnym pomocnikiem, który błyskawicznie wykonuje nawet najbardziej skomplikowane obliczenia.

Potrafi on również zarezerwować bilet lotniczy na dowolnej trasie, przesłać list, dowiedzieć się, co ciekawego wydarzyło się na świecie, a nawet zrobić zakupy. Nie musisz wcale wychodzić z domu ani z biura. Wszystkie informacje, nawet z bardzo odległych zakątków świata, mogą być przesyłane przez łącza telefoniczne.

Na świecie powstało już kilkanaście sieci wyspecjalizowanych w wymianie informacji komputerowych. Abonent korzysta z dostępu do ogromnych banków danych gromadzących wiedzę z różnych dziedzin nauki, bezpłatnego oprogramowania (public domain), serwisu prasowego i wielu innych usług. Jedną z takich sieci jest CompuServe, która działa w USA. Jej siedziba mieści się w Columbus w stanie Ohio. Obsługują ją potężne komputery VAX. Abonentem CompuServe może zostać użytkownik prawie każdego komputera: Atari 800XL, Atari ST, Commodore 64, Commodore Amiga, Apple II/III, Macintosh, IBM, TRS-80, Hewlett-Packard. Komputer wystarczy podłączyć do zwykłej linii telefonicznej przy pomocy modemu. Konstruktorzy modemów opracowali wiele interesujących rozwiązań. Niektóre z nich są bardzo proste, sprzęgają komputer z telefonem. Inne, inteligentne, potrafią same łączyć i odpowiadać na zgłoszenia użytkowników. Podstawowym parametrem, który decyduje o wyborze odpowiedniego modemu, jest szybkość transmisji danych wyrażona w bodach (bitach na sekundę). Najczęściej stosuje się modemy o szybkości 300 lub 1200 bodów. Do współpracy z siecią potrzebne jest jeszcze odpowiednie oprogramowanie. Na każdy z komputerów opracowano kilka pakietów oprogramowania komunikacyjnego, np. na Atari XL/XE programy 1030 1850 EXPRESS, HOMETERM, TSCOPE, SMART TERM, XETERM, PROXTERM. Część z nich rozpowszechniana jest bezpłatnie.

Korzystanie z sieci jest bardzo proste. Po połączeniu komputera z modemem i telefonem wystarczy uruchomić jeden z posiadanych pakietów komunikacyjnych. Następnie ustawiamy w programie odpowiednie parametry transmisji danych, które umożliwią naszemu komputerowi poprawne przekazywanie i odbiór informacji z sieci. Przesyłane polecenia, programy, teksty muszą być zapisane we wspólnym języku umożliwiającym odkodowanie przez dowolny komputer. Tym językiem jest kod ASCII, w którym każdej literze, cyfrze i innym znakom przyporządkowano jedną liczbę z zakresu 0-255. Niektóre specjalistyczne programy oraz pewne typy komputerów posługują się nieco odmiennym kodem (np. Atari kodem ATASCII). Należy więc pamiętać o ich przestawieniu na tryb ASCII. Kolejny istotny parametr to sposób, w jaki będzie prowadzona transmisja danych. Istnieją trzy możliwości: równoczesne przesyłanie informacji w obu kierunkach (dupleks), w jednym (simpleks) lub niejednoczesne w obu kierunkach (półdupleks). Komputer musi również w jakiś sposób wykrywać błędy w transmisji. Najczęściej stosowaną metodą jest kontrola parzystości. Polega to na dodaniu do każdego przekazywanego słowa tak dobranego bitu, by suma jedynek w tym słowie była zawsze parzysta. Odbiornik zliczając jedynki odebranego słowa łatwo może określić, czy któryś z bitów nie został przekłamany.

Po ustawieniu parametrów transmisji program komunikacyjny prosi o podanie numeru telefonicznego i łączy się z CompuServe. Uzyskanie dostępu do sieci jest uzależnione od wpisania prawidłowego numeru identyfikacyjnego użytkownika i hasła. System zabezpiecza się w ten sposób przed niepowołanymi osobami, które bezprawnie chciałyby korzystać z sieci. Hasło i numer identyfikacyjny są niezbędne również dla zagwarantowania ochrony informacji, które użytkownik sam przechowuje w sieci, np. poufne informacje o działalności przedsiębiorstwa. Istnieją również inne kryteria uprawniające do posługiwania się niektórymi danymi. Może to być przynależność do określonej grupy zawodowej, praca w firmie, itp. Niektóre dane, (np. systemowe) CompuServe *muszą być chronione przed niefachowym i lekkomyślnym użytkownikiem*. Ich uszkodzenie lub przypadkowe zniszczenie mogłoby spowodować znaczne szkody. Są więc udostępniane tylko przeszkolonym specjalistom.

Oczywiście z chwilą powstania sieci natychmiast pojawili się włamywacze, którzy usiłują złamać system zabezpieczeń, za darmo zdobyć programy, lub po prostu spłatać komuś psikusa. Poważnym problemem są również wirusy, które dzięki łączom telefonicznym mogą w błyskawicznym tempie rozprzestrzenić się na cały kontynent.

Każdy użytkownik, który choćby raz połączył się z siecią ma swój numer i kartotekę. Rejestrowane są tam wszystkie seanse łączności z CompuServe i na tej podstawie wyliczane opłaty. Podstawą rachunku jest czas połączenia, wielkość i czas przechowywania zbiorów w pamięci sieci oraz korzystanie z innych usług. Abonent reguluje rachunek w najdogodniejszej dla siebie formie. Może on zlecić automatyczne dokonywanie płatności przy pomocy karty kredytowej MasterCard, Visa lub American Express. Innym sposobem jest skorzystanie z komputerowej sieci CHECKFREE, która przelewa odpowiednią sumę na konto CompuServe bez konieczności podpisywania czeku.

Seans łączności z CompuServe zawsze rozpoczyna się od głównego menu. Użytkownik wybiera interesujący go temat i przechodzi do kolejnego, bardziej szczegółowego menu. Przypomina to wspinaczkę po pniu i gałęziach drzewa. Wszystkie przechowywane w sieci informacje są pogrupowane tematycznie. Użytkownicy pasjonujący się jakąś konkretną dziedziną mogą zrzeszać się w specjalnej grupie zainteresowania, zwanej forum. Właściciele 8 bitowych komputerów Atari tworzą forum Atari8. W obrębie forum można nawiązywać indywidualne kontakty. Do forum można dostać się wybierając kolejne opcje menu lub bezpośrednio, przy pomocy komendy GO (np. GO ATARI8). Lista możliwych GO obejmuje kilkadziesiąt pozycji. Wśród nich znajduje się wiele bardzo przydatnych w życiu codziennym.

Serwis informacyjny (GO NEWS). CompuServe dostarcza swoim abonentom informacje o aktualnych wydarzeniach na świecie, informacje gospodarcze, finansowe, sportowe i o pogodzie. Znajdują się tu doniesienia agencji Associated Press oraz artykuły z ważniejszych dzienników i magazynów w USA i Kanadzie.

Elektroniczne zakupy (GO SHOP). Niektóre renomowane firmy i domy towarowe oferują swoim klientom możliwość zrobienia zakupów za pośrednictwem sieci. Lista sprzedawanych w ten sposób wyrobów

Dokończenie na str. 31